AD/A-001 122

PARAMETRIC SCALING LAWS. PART II. A COMPUTER MODEL FOR DERIVING THE PERFORMANCE CHARACTERISTICS OF SATURATION-LIMITED PARAMETRIC SONARS

F. H. Fenlon, et al

Westinghouse Research Laboratories

Prepared for:

Office of Naval Research
Advanced Research Projects Agency

26 August 1974

AD A001122

PARAMETRIC SCALING LAWS

F. H. Fenlon and J. W. Kesner

Final Report, Part II - A003
"A Computer Model for Deriving the Performance Characteristics of Saturation-Limited Parametric Sonars"

ARPA Order Number:	2655/11-30-73
Program Code Number:	4E20
Effective Date of Contract:	February 1, 1974
Contract Expiration Date:	July 31, 1974
Amount of Contract:	$29,591
Contract Number:	N00014-74-C-0214
Principle Investigator:	F. H. Fenlon
	(412) 256-3676
Scientific Officer:	R. Obrochta

Sponsored by Advanced Research Projects Agency
ARPA Order No. 2655/11-30-73

This research was supported by the Advanced Research Projects Agency of the Department of Defense and was monitored by ONR under Contract No. N00014-74-C-0214

August 26, 1974

WESTINGHOUSE RESEARCH LABORATORIES PITTSBURGH, PA. 15235

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) PARAMETRIC SCALING LAWS - Part II - A Computer Model for Deriving the Performance Characteristics of Saturation-Limited Parametric Sonars		5. TYPE OF REPORT & PERIOD COVERED Final Report (Part II) 1 Feb. 1974 to 31 July 1974
		6. PERFORMING ORG. REPORT NUMBER 72-9M7-NONLN-R3
7. AUTHOR(s) F. H. Fenlon and J. W. Kesner		8. CONTRACT OR GRANT NUMBER(s) N00014-74-C-0214
9. PERFORMING ORGANIZATION NAME AND ADDRESS Westinghouse Electric Corporation Research & Development Center Beulah Road, Pittsburgh, PA 15235		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS ARPA Order No. 2655/11-30-73 Program Code No. 4E20
11. CONTROLLING OFFICE NAME AND ADDRESS Procuring Contracting Officer Office of Naval Research Department of the Navy Arlington, VA 22217		12. REPORT DATE August 26, 1974
		13. NUMBER OF PAGES 124
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Commander, Defense Contract Administration Services District 1610-S Federal Building, 1000 Liberty Ave. Pittsburgh, PA 15222		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

[redacted]

DISTRIBUTION STATEMENT A

Approved for public release; Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

NA

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Nonlinear, saturation, sonar

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report outlines the procedures used in a computer program for calculating the performance characteristics of saturation-limited parametric transmitters and receivers. The program enables the amplitude and frequency response of such sonars to be evaluated including the amplitude dependence of their far field beam patterns and directivity indices.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

PARAMETRIC SCALING LAWS

Part II: A Computer Model for Deriving the Performance Characteristics of Saturation-Limited Parametric Sonars

by

F. H. Fenlon and J. W. Kesner

SUMMARY

The purpose and significance of this ARPA sponsored report is to provide a detailed description and listing of the computer programs developed to compute the amplitude, frequency, and angular response characteristics of parametric sonars, from an analytical model developed in Part I. Technical results include examples of the performance characteristics and beam patterns generated by the program. Implications for further research would be to work backwards from the asymptotic far-field solutions derived in this report to compute the range dependence of the entire spectrum throughout the nonlinear interaction zone via the paraxial wave equation.

TABLE OF CONTENTS

LIST OF SYMBOLS i

THEORY 1

PROGRAM INPUT 7

PROGRAM OUTPUT 10

REFERENCES 11

APPENDIX A 13

FIGURE 1 (Pages 37-85) 37

FIGURE 2 (Pages 87-99) 87

FIGURE 3 (Pages 101-104) 101

PROGRAM LISTING (Pages 105-118) 105

Ib

LIST OF SYMBOLS

ρ_o	density of the medium
c_o	small signal speed of sound
$\beta = 1 + (B/2A)$	nonlinear parameter for liquids
$= \frac{1}{2}(1 + C_p/C_v)$	in gases
A, B	first and second-order coefficients in the equation of state
C_p, C_v	specific heats at constant pressure and constant volume
$r_o, r'_o = r_o(f_o/f_-)$	Rayleigh distance, and end-fire truncation distance
$f_i, (i=1,2)$	primary wave frequencies
$f_o = \frac{1}{2}(f_1+f_2)$	mean primary wave frequency
$f_\pm = f_1 \pm f_2$	sum or difference frequency
$k_i = 2\pi f_i/c_o, (i=1,2)$	primary wave numbers
$k_o = 2\pi f_o/c_o$	mean primary wave number
$\bar{p}_{oi}, (i=1,2)$	rms primary wave amplitudes at the source
$\bar{\delta} = \alpha/f^2$	nondispersive thermo viscous attenuation parameter
α	attenuation coefficient
$\alpha_i, (i=1,2)$	primary wave attenuation coefficients
$\alpha_\pm$	sum or difference-frequency attenuation coefficient
$\alpha_{T_\pm} = (\alpha_1+\alpha_2-\alpha_\pm)$	combined attenuation coefficient
$\alpha_T \equiv \alpha_{T_-}$	
α_o	mean primary wave attenuation coefficient
$\Gamma_o = \sigma_o/\alpha_o r_o$	mean primary wave acoustic Reynolds number

$X_i = \sigma_i E_1(\alpha_T r_o), (i=1,2)$

$X_o = \sigma_o E_1(\alpha_T r_o)$

$\sigma_i = \sqrt{2}\beta \bar{p}_{oi} k_i r_o / \rho_o c_o^2$

$\sigma_o = \beta \bar{p}_o r_o k_o / \rho_o c_o^2$

$\Lambda = E_1(\alpha_T r_o)\ e^{\alpha_T r_o}$

$\Lambda' = E_1(\alpha_T r_o')\ e^{\alpha_T r_o'}$

$$n = 1 + \frac{\log_{10}(\Lambda/\Lambda')}{\log_{10}(f_o/f_-)}$$

$E_1(z) = \int_z^\infty \frac{e^{-x}}{x}\,dx$	exponential integral
$D_i(\theta), (i=1,2)$	far-field primary wave directivity functions
$D_\pm(\theta)$	far-field sum or difference-frequency directivity function for a "spreading-loss-limited" parametric array
$\delta_\pm(\theta)$	far-field sum or difference-frequency directivity function for an "absorption-limited" parametric array
$b_\pm(\theta)$	convolved beam pattern
$d_\pm(\theta)$	$D_\pm(\theta)$, $\delta_\pm(\theta)$, or $b_\pm(\theta)$
DI	directivity index
SL_o	combined rms primary wave source level in dB//1μPam
SL_∞	equivalent mean rms primary wave source level at 1 m in dB//1μPam
$SL_\pm$	equivalent sum or difference-frequency rms source level at 1 m in dB//1μPam
$SL_o^* = SL_o + 20\log_{10} f_o$, ($f_o$ in kHz)	scaled combined rms primary wave source level in dB//1μPam kHz
$SL_\infty^* = SL_\infty + 20\log_{10} f_o$	scaled equivalent mean rms primary wave source level at 1 m in dB//1μPam kHz
$SL_\pm^* = SL_- + 20\log_{10} f_o$	scaled equivalent sum or difference frequency rms source level in dB//1μPa kHz
$20\log_{10}\eta_\pm = SL_\pm - SL_o$	parametric (sum or difference) conversion efficiency

THEORY

In Part 1 of this report an analytical solution for the far-field pressure of a parametric array was derived from Burgers' equation. Section 2 of Part I gave this solution in terms of scaled source levels. Since these scaled expressions form the basis of the computer program outlined in this part of the report, they are reexpressed here for the benefit of the reader as follows:

Let SL_o = the combined primary wave source level at 1m, in dB//1μPam

and $SL_\pm$ = the equivalent sum or difference-frequency source level referred to 1m, in dB//1μPam.

Then the scaled source levels SL_o^* and $SL_\pm^*$ are defined as,

$$\left.\begin{aligned} SL_o^* &= SL_o + 20\log_{10}f_o \quad &(1a)\\ SL_\pm^* &= SL_\pm + 20\log_{10}f_o \quad &(1b)\end{aligned}\right\} f_o \text{ in kHz}$$

where $$f_o = \tfrac{1}{2}(f_1 + f_2), \tag{2}$$

f_1 and f_2 being the operating frequencies of the parametric sonar. f_o is thus the mean primary wave frequency.

Parametric Transmitter:

From Section 2 of Part I the conversion efficiency η_- of a parametric transmitter can be expressed in logarithmic form as,

$$\begin{aligned} 20\log_{10}\eta_- &= (SL_- - SL_o)\\ &\equiv (SL_-^* - SL_o^*), \text{ from Eqs. (1a) and (1b)}\\ &= 20\log_{10}F - 20n\log_{10}(f_o/f_-) \qquad (3)\end{aligned}$$

where $$F = (2/x_o)\,\frac{I_1(x_1)I_1(x_2)}{I_o(x_1)I_o(x_2)}\;;\quad x_o = \sqrt{2}\,\bar{x}_o \tag{4}$$

and $$20\log_{10}x_o = SL_o^* + 20\log_{10}\Lambda + 20\log_{10}N \tag{5a}$$

$$20\log_{10}x_i = SL_i^* + 20\log_{10}\Lambda + 20\log_{10}N,\ (i=1,2) \tag{5b}$$

with $$N = (2\sqrt{2}\pi\ \beta/\rho_o c_o^3)(10^3) \tag{6}$$

and $$\Lambda = E_1(\alpha_T r_o)\exp(\alpha_T r_o);\quad \Lambda' = E_1(\alpha_T r_o')\exp(\alpha_T r_o');$$

$$\alpha_T r_o' = \alpha_T r_o(f_o/f_-) \tag{7}$$

$$n = 1 + \frac{\log_{10}(\Lambda/\Lambda')}{\log_{10}(f_o/f_-)} \tag{8}$$

The program computes, plots, and tabulates the conversion efficiency $20\log\eta_-$ as a function of the scaled combined primary wave source level SL_o^* for fixed values of $\alpha_T r_o$ and f_o/f_-. It also plots and tabulates the scaled difference-frequency source level SL_-^* as a function of SL_o^* for fixed values of $\alpha_T r_o$ and f_o/f_-. In addition the frequency response index n, as defined by Eq. 8 is tabulated. By reading in the relative amplitude $\bar{p}_{o1}/\bar{p}_{o2}$ of the rms primary wave amplitudes at the source the program will provide results for cases where the drive amplitudes are unequal.

Parametric Receiver:

For a limited class of problems such as that of a plane wave low frequency signal interacting with a plane wave "absorption limited" finite-amplitude source, or a spherical wave signal interacting with a spherical pump wave, the same expressions used to define the performance of a parametric transmitter will also define the performance of a parametric receiver if f_o/f_- is replaced by f_o/f_+ in Eq. 3 and if

$\alpha_T r_o$ is replaced by $-\alpha_T r_o$ in Eq. 7. The reason for this can be seen by writing $\alpha_T r_o$ as $\alpha_{T_\pm} r_o$ where,

$$\begin{aligned} \alpha_{T_\pm} r_o &= (\alpha_1 + \alpha_2 - \alpha_\pm) r_o \\ &= \bar{\delta}\{f_1^2 + f_2^2 - (f_1 \pm f_2)^2\} r_o \\ &= \mp 2\bar{\delta} f_1 f_2 r_o. \end{aligned} \tag{9}$$

Thus $\alpha_{T_-} r_o = 2\bar{\delta} f_1 f_2 r_o$, for the difference-frequency signal (10a)

and $\alpha_{T_+} r_o = -2\bar{\delta} f_1 f_2 r_o$
$= -\alpha_{T_-} r_o$, for the sum-frequency signal. (10b)

In the computer output $20 \log_{10} \eta_-$ becomes $20 \log_{10} \eta_+$ and SL^*_- becomes SL^*_+. The substitution of $z = -\alpha_T r_o$ in the exponential integral $E_1(z)$ transforms this to $E_1(z)$, as defined by Abramowitz and Segun.[1]

It should be noted in the case of a parametric receiver that the index n is still calculated by Eq. 8 since the sum frequency component must have the same frequency response as the difference-frequency component.

Evaluation of $E_1(z)\exp(z)$ for $z \geqslant 10.0$:

For $z \geqslant 10.0$, the function $E_1(z)\exp(z)$ which is used in calculating ℓ, ℓ', and ℓ_o as defined by Eqs. 7 and 22, respectively, is determined from an approximate expression given by Abramowitz and Segun[1] as,

$$E_1(z)\exp(z) = \frac{z^2 + A_1 z + A_2}{z^3 + B_1 z^2 + B_2 z}$$

where $A_1 = 4.03640$

$A_2 = 1.15198$

$B_1 = 5.03637$

$B_2 = 4.19160$

Far-Field Directivity Function:

For a "spreading-loss-limited" parametric array ($\alpha_T r_o' \leq 10^{-2}$ dB) the program gives the far-field directivity function $D_{\pm}(\theta)$ as,

$$D_{\pm}(\theta) = \left\{\frac{I_1[\chi_1 D_1(\theta)]\, I_1[\chi_2 D_2(\theta)]}{I_o[\chi_1 D_1(\theta)]\, I_o[\chi_2 D_2(\theta)]}\right\} \Big/ \left\{\frac{I_1(\chi_1)\, I_1(\chi_2)}{I_o(\chi_1)\, I_o(\chi_2)}\right\}, \tag{11}$$

where $D_i(\theta)$, $(i=1,2)$ are the far-field primary wave directivity functions given in the program as,

$$D_i(\theta) = \frac{2J_1(k_i a \sin\theta)}{k_i a \sin\theta}; \ (i=1,2), \text{ for a circular piston of radius } a \tag{12}$$

$$= \frac{\sin(k_i a \sin\theta)}{k_i a \sin\theta}; \ (i=1,2), \text{ for a line array of length } 2a. \tag{13}$$

It should be noted that although Eq. 11 is only applicable for $\alpha_T r_o' \leq 10^{-2}$ dB, the program permits it to be evaluated for all values of $\alpha_T r_o'$. For values of $\alpha_T r_o' > 10^{-2}$ dB therefore, the "absorption-limited," "virtual-end-fire-array" directivity function $\delta_{\pm}(\theta)$ should be used instead. This function is given in the program by the expression,

$$\delta_{\pm}(\theta) = \left\{\sum_{n=o}^{\infty} \frac{a_n}{(n+1)\alpha_{T_\pm} r_o + j2k_{\pm} r_o \sin^2(\theta/2)}\right\} \Big/ \left\{\sum_{n=o}^{\infty} \frac{a_n}{(n+1)\alpha_{T_\pm} r_o}\right\} \tag{14}$$

where $$a_n = (\Gamma_o/r)^n \frac{\sin[(n+1)\tan^{-1}(4/\Gamma_o)]}{[\sqrt{1 + (\Gamma_o/4)^2}]^{n+1}} \tag{15}$$

and $$20 \log_{10} \Gamma_o = SL_o^* - 20 \log_{10}\left|\frac{\alpha_T r_o}{2}\right| + 20 \log_{10} N; \; \bar{p}_{o1} = \bar{p}_{o2}. \tag{16}$$

It should be noted that unlike Eq. 11, Eq. 14 only holds for equal primary wave amplitudes. Although it resembles Bartram's[2] end-fire-directivity function, Eq. 14 is a new and more general form of the directivity function described by Childs.[3] It was derived from Kuznetsov's equation[4] via Burgers' equation and will be the subject of a later article.

For $10^{-2} \lesssim \alpha_T r_o' \lesssim 10$ the convolution of Eqs. 11 and 14, which is available in the program, should be used to obtain the far-field parametric array beam pattern $b_{\pm}(\theta)$ where,

$$b_{\pm}(\theta) = \int_{-\pi/2}^{\pi/2} D_{\pm}(\theta')\delta_{\pm}(\theta - \theta')\cos\theta' \, d\theta' \tag{17}$$

where $D_{\pm}(\theta)$ is defined by Eq. 11 and $\delta_{\pm}(\theta)$ is given in Eq. 14.

Directivity Index:

The directivity index DI is calculated in the program as a function of the scaled source level SL_o^* for fixed values of $\alpha_T r_o$ and f_o/f_-, from the expression,

$$DI = 10 \log_{10} \frac{2}{\int_0^{\pi} \left| d_{\pm}(\theta') d_{\pm}^*(\theta') \right| \sin\theta' \, d\theta'} \tag{18}$$

where $d_{\pm}(\theta)$ stands for $D_{\pm}(\theta)$, $\delta_{\pm}(\theta)$, or $b_{\pm}(\theta)$ whichever is appropriate and $d_{\pm}^*(\theta)$ is its complex conjugate.

Far-Field Monofrequency Source Level:

The program can also be used to obtain the referred far-field scaled source level at 1 m, SL_∞^* of a monofrequency source of frequency f_o, as a function of the scaled "input" source level SL_o^*, for fixed values of $\alpha_o r_o$,

where $$SL_{\infty}^{*} = SL_{o}^{*} + 20 \log_{10} F' \tag{19}$$

and $$F' = (2/\chi_o)\frac{I_1(\chi_o)}{I_o(\chi_o)} \tag{20}$$

with $$20 \log_{10} \chi_o = SL_o^{*} + 20 \log_{10} \Delta_o + 20 \log_{10} N \tag{21}$$

$$\Delta_o = E_1(2\alpha_o r_o)\exp(2\alpha_o r_o). \tag{22}$$

Eqs. 19 to 22 will be the subject of a later article. Note that

$$SL_{\infty}^{*} = SL_{\infty} + 20 \log_{10} f_o; \quad (f_o \text{ in kHz}), \tag{23}$$

where SL_∞ will differ from SL_o as the latter increases, due to the transfer of energy from f_o into self-generated harmonics in the medium. In the case of a parametric source the program gives SL_{∞}^{*} vs SL_o^{*} as an "input-output" relationship for the mean primary wave frequency.

PROGRAM INPUT

The data for the program is in free field format which means that the data is not restricted to any particular columns of the data card. Input variables which begin with the letters I, J, K, L, M, N are integers (no decimal point). All others are real variables and must have a decimal point. The data values on a card must be separated from each other by commas.

Card 1 NOC = the number of separate cases being run.

Card 2 IOP, IOE, IOS, IOD, IOO

Each of these integer indicators must be either 0 or 1. A zero indicates that that particular part of the program can be skipped. A one indicates that that section is to be done. IOP is for the pattern section. IOE is for the 20 log (ETA) section. IOS is for the SPL-* section. IOD is for the directivity index section. IOO is for the SPL (OUT) section.

Card 3 ALTRO, P1P2

ALTRO = $\alpha_T r_o$ in dB. ALTRO is positive for difference-frequency cases and is negative for sum frequency cases. P1P2 is the ratio (P1/P2) of the primary wave pressure amplitudes of the two drive frequencies F1 and F2 at the source.

Card 4 NFM, NFMRD

NFM = the number of FO/F- values. NFM must be 5 or less.

NFMRD = 0 or 1. A zero will generate a set of FO/F- cases according to the formula FO/F- = $5.0*2^{K-1}$, K=1,..., NFM. If NFMRD = 1 then NFM values for FO/F- must be read in.

Card 5 FFM(K), K = 1, . . . , NFM

FFM(K) = the K^{th} value of FO/F-. If NFMRD = 0 this card must be omitted.

Card 6 LIN, IOC

LIN = 0 for a piston, = 1 for a rectangle.

IOC = 1 for the "I" Bessel function formula for the beam pattern.

= 2 for the beam pattern of the delta formula.

= 3 for the beam pattern of the convolution of the delta formula with the "I" Bessel function formula. If IOP = 0 and IOD = 0 then this card must be omitted.

Card 7 AKOR if LIN = 0

AKOL, AKOW if LIN = 1

AKOR = $k_o a$ = 2.0 * π * piston radius/wavelength.

AKOL = $k_o \ell$ = 2.0 * π * rectangle length/wavelength.

AKOW = $k_o w$ = 2.0 * π * rectangle width/wavelength.

The wavelength is of FO (FO = (F1+F2)/2.0). If IOP = 0 and IOD = 0 then this card must be omitted.

Card 8 SLS, PFFM, DB, NP, TD

SLS = the value in dB of SLO* at which a pattern is desired.

PFFM = the value of FO/F- at which a pattern is desired.

DB = the lower limit of the pattern plot (e.g. DB = -80.0)

NF = the number of theta values at which the beam pattern is evaluated.

TD = $\Delta\theta$ = the step size between theta values in degrees.

If IOP = 0 this card must be omitted.

Some of the sections have graphs with multiple curves. The program presently limits the number of curves per graph to five.

The graphs that have SLO* as the X-axis are presently limited to twenty-one (21) separate values of SLO*. These are generated in the program by the formula SLO* = 180.0 + 10.0 * (J-1), J = 1, 2, 3, . . . , 21.

The maximum number of theta values for patterns is restricted to 101 because of the size of the plotting grid.

If a pattern is desired from a rectangular source the pattern is assumed to be in the plane of the length axis of the rectangle.

For the directivity index of a piston the pattern is assumed to be zero behind the piston.

Three subroutines are used in the program--BMPAT, DLPAT, CNPAT.

BMPAT is used to calculate the beam pattern for IOC = 1.

DLPAT is used to calculate the beam pattern for IOC = 2.

CNPAT is used to calculate the beam pattern for IOC = 3.

The program calls on several packaged routines to evaluate one-dimensional definite integrals, N-dimensional definite integrals, "I" Bessel functions, "J" Bessel functions, and exponential integrals. Packaged plotting routines are used as well as a routine to determine the amount of CPU time used by the program. All of these routines are described in Appendix A.

PROGRAM OUTPUT

Figure 1 shows an example of the program output for 20 log(ETA), SPL-*, and SPL(OUT) for a range of $\alpha_T r_0$ values from 10^{-5} dB to 10^{2} dB. Five values of FO/F- were used varying from 5.0 to 80.0.

Figure 2 shows an example of the program output for the patterns of a piston with a k_0a of 10.0 where FO/F- = 10.0 and SPLO* is varied from 240 dB to 330 dB.

Figure 3 shows the program output for the directivity index of a piston with a k_0a of 10.0 and an FO/F- of 10.0.

REFERENCES

1. M. Abramowitz and I. A. Segun, "Handbook of Mathematical Functions," (Dover Publications, Inc. 1965) pp. 228-237.
2. J. F. Bartram, "A Useful Analytical Model for the Parametric Acoustic Array," J. Acoust. Soc. Am. 52, 1042-1044 (1972).
3. D. R. Childs, "Beam Patterns and Directivity Indices of Parametric Acoustic Arrays," Proc. Symp. Finite-Amplitude Wave Effects in Fluids, Technical University of Denmark, Copenhagen (August 1973).

(This page intentionallv left blank.)

APPENDIX A

Included in this appendix are descriptions of the packaged routines that are used in the program. The routines ROMBS, ROM2, RMB1, RMB2, RMB3, BJY01, BESI, and DEI are from the JPL Fortran V Subroutine Directory, Edition No. 4, October 1970. The routines PSCALE, PSETUP, PPLOT, and SECOND are from the Westinghouse Research Laboratories' Fortran library.

11.1. ONE-DIMENSIONAL

11.1.1. QUADRATURE, ONE-DIMENSION, S.P.

11.1.1.1. IDENTIFICATION

QUADRATURE, ONE-DIMENSIONAL, SINGLE PRECISION

LANG	FILE	ELT/VERS	SIZE	ENTRY NAMES
F-V	LIB*JPL$	ROMBS/JPL	1251(8)=681(10)	ROMBS,ROM2

SUBROUTINES USED: #NONE#

COGNIZANT PERSONS: W. R. BUNTON AND M. DIETHELM,
JPL, SECTION 314, 1969 SEPT 30

11.1.1.2. PURPOSE

OBTAIN APPROXIMATE EVALUATION OF A ONE-DIMENSIONAL DEFINITE INTEGRAL BY NUMERICAL QUADRATURE.

ANS = THE INTEGRAL FROM A TO B OF F(X)*DX

11.1.1.3. REFERENCES:

FOR A COMPLETE DESCRIPTION OF THIS SUBROUTINE, INCLUDING A DISCUSSION OF A VARIETY OF TEST CASES, SEE:

WILEY R. BUNTON, MICHAEL DIETHELM, AND KAREN HAIGLER, 'ROMBERG QUADRATURE SUBROUTINE FOR SINGLE AND MULTIPLE INTEGRALS', JPL

W. BUNTON, M. DIETHELM, G. WINJE, 'MODIFIED ROMBERG QUADRATURE: A SUBROUTINE TO SUPPORT GENERAL SCIENTIFIC COMPUTING', JPL INTERNAL MEMORANDUM TM 314-208, APRIL 1, 1970.

W. BUNTON, M. DIETHELM, 'MODIFICATIONS TO THE JPL ROMBERG SUBROUTINES', TM 314-247, 1 SEPT. 1970.

11.1.1.4. METHOD

THIS SUBROUTINE COMBINES TECHNIQUES FROM 'ROMBERG' AND 'ADAPTIVE STEP' QUADRATURE METHODS. THE SUBROUTINE INITIALLY PICKS A SUBINTERVAL [A,B1] OF THE TOTAL INTERVAL [A,B] AND ATTEMPTS TO APPROXIMATE THE INTEGRAL OVER THIS SUBINTERVAL BY USING THREE

STAGES OF ROMBERG QUADRATURE. THIS REQUIRES EVALUATION OF THE INTEGRAND AT FIVE EQUALLY SPACED POINTS.

IF THIS QUADRATURE IS REGARDED AS SUCCESSFUL (SEE: ERROR CONTROL) THE SUBROUTINE WILL ADD THE VALUE OBTAINED TO A RUNNING SUM AND PROCEED TO TREAT A NEW DISJOINT SUBINTERVAL OF THE SAME OR GREATER LENGTH.

IF THIS QUADRATURE IS NOT REGARDED AS SUCCESSFUL AND THE CURRENT STEP LENGTH IS GREATER THAN HMIN, THEN THE SUBROUTINE WILL REJECT THE RESULT FOR THE CURRENT SUBINTERVAL AND TAKE THE LEFT HALF OF THE SUBINTERVAL AS THE NEW SUBINTERVAL TO BE TREATED.

IF THE CURRENT STEP LENGTH IS HMIN OR SMALLER, THEN THE SUBROUTINE WILL ACCEPT THE CURRENT RESULT AND PROCEED TO THE NEXT SUBINTERVAL, WRITING A MESSAGE ON FORTRAN UNIT 6 IDENTIFYING THE SUBINTERVAL ON WHICH THE ACCURACY TEST WAS NOT SATISFIED.

11.1.1.5. ERROR CONTROL

THE QUADRATURE IS REGARDED AS SATISFACTORY OVER A PARTICULAR SUBINTERVAL IF

1. THE ESTIMATED RELATIVE ERROR OF THE QUADRATURE OVER THAT SUBINTERVAL IS AT MOST ERMAX, OR

2. THE ESTIMATED ERROR IN THAT SUBINTERVAL RELATIVE TO THE ACCUMULATED VALUE OF THE INTEGRAL UP TO AND INCLUDING THAT SUBINTERVAL IS AT MOST ERMAX.

3. THE ESTIMATED ABSOLUTE ERROR OVER THAT SUBINTERVAL IS AT MOST .1*ABS(ERMAX).

11.1.1.6. PARAMETER CHECKING

THE SUBROUTINE WILL TERMINATE EXECUTION WITH A PRINTED MESSAGE AND A STANDARD EXEC 8 WALK-BACK (RETURN 0) IF THE GIVEN PARAMETERS DO NOT SATISFY

A<B, OR A>B, BUT NOT A=B
0.<HMIN.LE.HSTAR.LE.HMAX AND

RELATIVE - 0.<ERMAX<1.0, BUT NOT ERMAX=0.
ABSOLUTE - ANY NEGATIVE NUMBER, BUT NOT =0.

IF A.GT.B OR IF HSTAR.GE.(B-A)/4, ONLY A WARNING MESSAGE IS PRINTED.

11.1.1.7. USAGE

```
      REAL      A, B, X, FOFX, HSTAR, HMIN, HMAX, ERMAX, ANS
      INTEGER   K, KEY

      [ASSIGN VALUES TO A, B, HSTAR, HMIN, HMAX, ERMAX, AND KEY]

      CALL  ROMBS(A,B,X,FOFX,HSTAR,HMIN,HMAX,ERMAX,ANS,K,KEY)

   10 [EVALUATE THE INTEGRAND USING THE CURRENT VALUE OF  X  AND
      STORE THE RESULT IN  FOFX.]

      CALL  ROM2

      IF(K.EQ.1) GO TO 10
      [QUADRATURE IS COMPLETED.  RESULT IS IN  ANS]
```

THE SUBROUTINE PARAMETERS ARE DEFINED AS FOLLOWS:

A,B LIMITS OF INTEGRATION. REQUIRE A NOT=B.

X VARIABLE SET BY THE SUBROUTINE FOR INTEGRAND EVALUATION IN THE USER'S PROGRAM.

FOFX VALUE OF INTEGRAND COMPUTED BY USER'S PROGRAM USING THE ARGUMENT X.

HSTAR SUGGESTED INITIAL STEP SIZE. THE INITIAL STEP SIZE, H, WILL BE SET AT H=.01*(B-A) IF HSTAR.GE.(B-A)/4. OR AT H=HSTAR IF HSTAR.LT.(B-A)/4. THE FIRST SUBINTERVAL WILL BE OF LENGTH 4.*H AND WILL REQUIRE FIVE EVALUATIONS OF THE INTEGRAND AT X=A,A+H,...,A+4*H. SUGGEST HSTAR.GE.(B-A)/4.

REQUIRE HMIN .LE. HSTAR .LE. HMAX.

HMIN MINIMUM ALLOWABLE STEP SIZE. REQUIRE 0. < HMIN .LE. HSTAR

HMAX MAXIMUM ALLOWABLE STEP SIZE. REQUIRE HSTAR .LE. HMAX.

ERMAX TOLERANCE ON RELATIVE OF ABSOLUTE ERROR. SEE DISCUSSION ABOVE UNDER 'ERROR CONTROL'. REASONABLE SETTINGS FOR ERMAX WOULD BE IN THE RANGE FROM 1.E-4 TO 1.E-7. IF GREATER ACCURACY IS REQUIRED, SEE THE WRITE-UP ON ROMBD. IT IS REQUIRED THAT 0.LT.ERMX.LT.1. FOR THE RELATIVE ERROR TEST. ONE SHOULD KNOW THE RANGE OF THE ANSWER

BEFORE HE USES ABSOLUTE ERROR.

ANS THE FINAL VALUE OF THE INTEGRAL. AVAILABLE WHEN ROM2 RETURNS WITH K=2.

K BRANCHING FLAG SET BY THE SUBROUTINE FOR USE IN THE USER'S PROGRAM. K=1 MEANS THE USER SHOULD EVALUATE THE INTEGRAND AT X, STORE THE VALUE IN FOFX, AND RE-ENTER ROM2. K=2 MEANS THE COMPUTATION IS COMPLETED AND THE VALUE IS IN ANS.

KEY FLAG TO CONTROL PRINTING OF ERROR MESSAGES. PREVIOUSLY, ANY VALUE OF KEY NOT = 7 WOULD WRITE A DIAGNOSTIC MESSAGE WHEN H BECAME.LT.HMIN. THE INPUT VALUES HAVE BEEN CHANGED SO THAT WHEN

	ACTION
KEY=5	PRINT INTERMEDIATE T AND Y VALUES; PRINT THE HMIN DIAGNOSTIC IF DETECTED.
=6	PRINT INTERMEDIATE T AND Y VALUES; DO NOT PRINT THE HMIN DIAGNOSTIC.
=7	DO NOT PRINT THE T AND Y VALUES OR HMIN DIAGNOSTIC.
=ANY OTHER VALUE	PRINT THE HMIN DIAGNOSTIC IF DETECTED.

THE T VALUES PRINTED ARE T(1,0), T(1,1), AND T(2,0). THE PRINTED Y VALUES ARE THE FUNCTIONAL EVALUATIONS Y(1) THRU Y(5) AT THE POINTS X,X+H,...,X+4H. SEE REFERENCES.

11.1.1.8. REMARKS:

THE FOLLOWING NOTES MAY BE APPLICABLE FOR DIFFICULT INTEGRANDS.

1. IF IN DOUBT, ONE SHOULD USE A SMALL VALUE OF HSTAR. THE STEP SIZE H CAN DOUBLE QUICKLY AND THE USER IS PENALIZED ONLY A SMALL NUMBER ON FUNCTIONAL EVALUATIONS WHILE HE INCREASES HIS CHANCES OF GETTING AN ACCURATE APPROXIMATION MANY FOLD.

2. BE CAUTIOUS WHEN RELATIVE ERMAX IS .GT. 10-5. IF HSTAR.LT.(B-A)/4, BUT NOT SMALL ENOUGH, AND THE FUNCTION IS OCCILATORY, VERY DIFFICULT, ETC., ROMBS CAN RETURN A WRONG ANSWER. EXAMPLE: F(X)=X*SIN30X*COSX ON THE INTERVAL (0,2PI), HSTAR=1.57, TRUE ANSWER=-.20967248, A RELATIVE TOLERANCE OF .1 GAVE -.25E-5, AND A RELATIVE TOLERANCE OF .01 GAVE 4.188; BAD ANSWERS.

3. ALSO FOR RELATIVE ERMAX .LT.10**-6. IF THE RELATIVE ERROR IS ASKING FAR GREATER THAN 6 SIGNIFICANT DIGITS ONE IS PUSHING THE ACCURACY OF THE UNIVAC 1108. ON A HIGHLY OCCILATORY, VERY DIFFICULT PROBLEM, ROMBS MAY BE TAKING THOUSANDS MORE FUNCTIONAL EVALUATIONS AND NOT REACHING THE ACCURACY IT DID AT 10**-6. EXAMPLE: SAME F(X) AS ABOVE WHEN GIVEN TOLERANCE WAS 10**-6, ANS=.20967380 WITH A 2009 FUNCTIONAL EVAUATIONS; BUT WHEN WHEN ERMAX=10**-8, ANS=.20967765 AND 10,267 FUNCTIONAL EVALUATIONS.

4. IF ONE WANTS A ROUGH APPROXIMATION OF THE INTRGRAL, HE CAN SET HMIN=HSTAR=HMAX. A FIXED STEP INTRGRATION OF THE FUNCTION WILL TAKE PLACE. BE SURE THAT KEY=7, OR MANY DIAGNOSTICS MAY BE PRINTED.

11.2. MULTI-DIMENSIONAL

11.2.1. QUADRATURE, MULTI-DIMENSION, S.P.

11.2.1.1. IDENTIFICATION

QUADRATURE, MULTI-DIMENSIONAL, SINGLE PRECISION

LANG	FILE	ELT/VERS	SIZE	ENTRY NAME
F-V	LIB*JPL$	RMB1/JPL	361(8)=241(10)	RMB1
F-V	LIB*JPL$	RMB/JPL	2267(8)=1207(10)	RMB1A,RMB2 RMB3

SUBROUTINES USED: #NONE#

COGNIZANT PERSONS: W. R. BUNTON AND M. DIETHELM,
JPL, SECTION 314, 1969 SEPT 30

```
- - - - - - - - - - - - - - - - - - - - - - - - -
-  CALLING SEQUENCE TO THIS SUBROUTINE        -
-  HAS BEEN CHANGED SINCE ED. 3 1 APRIL 70-
-  VERSION OF THIS DIRECTORY.                    -
- - - - - - - - - - - - - - - - - - - - - - - - -
```

11.2.1.2. PURPOSE

OBTAIN APPROXIMATE EVALUATION OF AN N-DIMENSIONAL DEFINITE INTEGRAL BY NUMERICAL QUADRATURE. THE LIMITS OF INNER INTEGRALS CAN BE FUNCTIONS OF THE VARIABLES OF THE OUTER INTEGRALS.

RESULT = THE INTEGRAL FROM A1 TO B1
WITH RESPECT TO X(1) OF

[THE INTEGRAL FROM A2 TO B2 WITH RESPECT TO X(2) OF
[...[THE INTEGRAL FROM AN TO BN WITH RESPECT TO X(N) OF F]
...]]

WHERE

1. A1 AND B1 ARE CONSTANTS

2. FOR J=2,...,N, AJ AND BJ MAY BE FUNCTIONS OF X(1),...,X(J-1).

3. THE INTEGRAND F, IS A FUNCTION OF X(1),...,X(N).

11.2.1.3. REFERENCES

FOR A COMPLETE DESCRIPTION OF THIS SUBROUTINE, INCLUDING A DISCUSSION OF A VARIETY OF TEST CASES, SEE:

WILEY R. BUNTON, MICHAEL DIETHELM, AND KAREN HAIGLER, 'ROMBERG QUADRATURE SUBROUTINE FOR SINGLE AND MULTIPLE INTEGRALS', JPL INTERNAL MEMORANDUM TM 314-221, JULY 1, 1969. SEE OTHER REFERENCES UNDER ROMBS.

11.2.1.4. METHOD

THE SUBROUTINE PERFORMS A NESTED SEQUENCE OF ONE-DIMENSIONAL INTEGRATIONS. THE SUBROUTINE USED FOR THE ONE-DIMENSIONAL INTEGRATIONS IS ESSENTIALLY ROMBS WITH CHANGES IN THE WAY STORAGE IS MANAGED. SEE THE WRITE-UP ON ROMBS.

11.2.1.5. USAGE

THE USER MUST PROGRAM AN INITIALIZATION CALL TO RMB1, THEN N SEPARATE CALLS TO RMB2, ONE FOR EACH LEVEL OF INTEGRATION, AND FINALLY A CALL TO RMB3. SOME OF THESE CALL STATEMENTS WILL BE EXECUTED MORE THAN ONCE.

THE CODING SPECIFICATIONS ARE AS FOLLOWS:

```
      INTEGER N,KGO,KEY
      REAL    X(NN1N), F, RESULT, ERMAX, W(NN2N)
      REAL    A1, B1, HSTAR1, HMIN1, HMAX1
        .       .
        .       .
        .       .
      REAL    AN, BN, HSTARN, HMINN, HMAXN
```

WARNING: CALLING SEQUENCE TO THIS SUBROUTINE HAS BEEN CHANGED SINCE ED. 3, 1 APRIL 1970 VERSION OF THIS DIRECTORY.

```
         [ASSIGN VALUES TO  N  AND  ERMAX.]
      CALL RMB1(N,X,F,RESULT,ERMAX,KGO,W,KEY)

    1     [ASSIGN VALUES TO ALL PARAMETERS IN THE FOLLOWING
           CALL  STATEMENT.]
      CALL  RMB2(A1,B1,HSTAR1,HMIN1,HMAX1)

    2     [ASSIGN VALUES TO ALL PARAMETERS IN THE FOLLOWING
           CALL  STATEMENT.  SOME OF THESE VALUES MAY BE COMPUTED
           AS FUNCTIONS OF  X(1).]
```

```
      CALL  RMB2(A2,B2,HSTAR2,HMIN2,HMAX2)
      .
      .
      .
 #N#     [ASSIGN VALUES TO ALL PARAMETERS IN THE FOLLOWING
          CALL  STATEMENT.  SOME OF THE VALUES MAY BE COMPUTED AS
          FUNCTIONS OF  X(1),...,X(N-1).]
      CALL  RMB2(AN,BN,HSTARN,HMINN,HMAXN)

 #N+1#   [COMPUTE  F  AS A FUNCTION OF  X(1,...,X(N).]
      CALL  RMB3
      GO TO  (1,2,...,#N#,#N+1#,#N+2#),KGO

 #N+2#   [THE COMPUTATION IS COMPLETED.  THE VALUE OF THE INTEGRAL
          IS IN  RESULT.]
```

THE DIMENSIONING PARAMETERS MUST SATISFY:

```
          #N1# .GE. N
          #N2# .GE. 29*N
```

THE PARAMETERS FOR RMB1 ARE DEFINED AS FOLLOWS:

N — NUMBER OF LEVELS OF INTEGRATION REQUIRED. (N .GE. 1). EXAMPLE: FOR A TRIPLE INTEGRAL, SET N=3.

(X(I)I=1,N) — CURRENT VALUES OF THE INTEGRATION VARIABLES SET BY THE SUBROUTINE. X(1) IS ASSOCIATED WITH THE OUTERMOST INTEGRATION AND X(N) WITH THE INNERMOST.

F — VALUE OF INTEGRAND, TO BE COMPUTED BY THE USER AS A FUNCTION OF X(1),...,X(N) AFTER THE N-TH CALL TO RMB2 AND WHEN RMB3 RETURNS WITH KGO = N+1.

RESULT — VALUE OF INTEGRAL COMPUTED BY THE SUBROUTINE. AVAILABLE WHEN RMB3 RETURNS WITH KGO = N+2.

ERMAX — RELATIVE OR ABSOLUTE ERROR TOLERANCE SET BY THE USER. REASONABLE VALUES WOULD BE IN THE RANGE 1.E-2 TO 1.E-4 FOR RELATIVE. REQUESTING MORE ACCURACY MAY GREATLY INCREASE THE EXECUTION TIME. ERMAX MUST SATISFY 0.LT.ERMAX.LT.1. FOR RELATIVE, ANY NEGATIVE NUMBER NOT = 0 FOR ABSOLUTE.

KGO — BRANCHING FLAG SET BY THE SUBROUTINE RMB3 FOR USE IN THE USER'S PROGRAM.

KGO = 1,...,N MEANS THE USER SHOULD SET THE PARAMETERS FOR THE KGO-TH CALL TO RMB2 AND THEN EXECUTE THAT

CALL.

KGO = N+1 MEANS THE USER SHOULD COMPUTE THE INTEGRAND, F, AND THEN CALL RMB3.

KGO = N+2 MEANS THE COMPUTATION IS COMPLETED AND THE VALUE OF THE INTEGRAL IS IN RESULT.

(W(I),I=1,29*N) WORKING SPACE NEEDED BY THE SUBROUTINE.

KEY FLAG TO CONTROL PRINTING OF ERROR MESSAGE WHEN ERROR TOLERANCE IS NOT MET WITH MINIMUM STEP SIZE.

KEY=7 NO ERROR MESSAGE IS PRINTED
KEY=ANY OTHER VALUE ERROR MESSAGE IS PRINTED.

THE PARAMETERS FOR THE J-TH CALL TO RMB2 ARE AS FOLLOWS:

AJ,BJ LOWER AND UPPER LIMITS OF INTEGRATION FOR THE J-TH INTEGRAL. EITHER AJ > BJ OR AJ .LE. BJ IS PERMITTED.

HSTARJ,HMINJ,HMAXJ QUADRATURE STEP PARAMETERS FOR THE J-TH INTEGRAL. THESE PARAMETERS MUST BE POSITIVE, REGARDLESS OF THE SIGN OF (BJ-AJ), AND MUST SATISFY 0. < HMINJ .LE. HSTARJ .LE. HMAXJ. THE J-TH QUADRATURE WILL START WITH A STEP WHOSE MAGNITUDE IS

HJ=MIN(HSTARJ,ABS(BJ-AJ)/4.) EXCEPT FOR OUTER INTEGRAL. SEE REMARK 4. BELOW.

AND THE MAGNITUDE OF THE STEP WILL BE KEPT BETWEEN HMINJ AND HMAXJ.

11.2.1.6. REMARKS

1. THE VALUE OF KGO WILL NEVER BE 1, THUS THE FIRST CALL TO RMB2 WILL ONLY BE EXECUTED ONCE.

2. THE PARAMETERS N, ERMAX, AND ALL PARAMETERS IN THE CALLS TO RMB2 WHICH ARE TO REMAIN CONSTANT DURING THE QUADRATURE CAN BE SPECIFIED LITERALLY IN THE CALL STATEMENTS.

3. PARAMETERS IN THE SECOND THROUGH THE N-TH CALL TO RMB2 WHICH ARE NOT CONSTANT CAN BE WRITTEN IN THE CALL STATEMENTS AS ARITHMETIC EXPRESSIONS. THE PARAMETER F, HOWEVER, CANNOT BE REPLACED IN THE CALL TO RMB1 BY AN ARITHMETIC EXPRESSION EXCEPT FOR THE SPECIAL CASE IN WHICH F IS CONSTANT.

4. IF THE STARTING STEP SIZE FOR THE OUTER INTEGRAL IS GREATER

THAN OR EQUAL TO (B-A)/4., A WALK-BACK DIAGNOSTIC (RETURN 0) IS PRINTED AND EXECUTION IS TERMINATED.

5. SUGGESTED VALUE FOR KEY IS 0.

11.2.1.7. EXAMPLE

CASE 11, PAGE 48, OF THE REFERENCE CITED ABOVE CAN BE CODED AS FOLLOWS :

```
      REAL  X(3), WORK(87)

      CALL RMB1(3,X,F,RESULT,1.E-4,KGO,WORK,0)
   10 CALL  RMB2(0.        , 1.        , 1.E-2, 1.E-4, 1.)
   20 CALL  RMB2(X(1)      , X(1)**2   , 1.E-2, 1.E-4, 1.)
   30 CALL  RMB2(X(1)*X(2), X(1)+X(2), 1.E-2, 1.E-4, 1.)
   40 F = X(1)*X(2)*X(3)
      CALL  RMB3
      GO TO (10, 20, 30, 40, 50), KGO
   50 CONTINUE
```

THE TRUE VALUE OF THIS INTEGRAL IS -0.032060185 . THE COMPUTED VALUE HAD AN ABSOLUTE ERROR OF 5.1E-9 AND A RELATIVE ERROR OF 1.6E-7 . THE SUBROUTINE WROTE TWO DIAGNOSTIC MESSAGES INDICATING THAT THE REQUESTED ACCURACY COULD NOT BE ATTAINED WITH THE MINIMUM STEP SIZE ON THE INTERVALS 0.0 .LE. X(1) .LE. 4.E-4 AND 4.E-4 .LE. X(1) .LE. 8.E-4 . THIS EXAMPLE USED 10361 EVALUATIONS OF THE INTEGRAND.

THIS EXAMPLE WAS RUN 20 TIMES AND TIMED USING PRTIM1/PRTIM2. THE TIME VARIED FROM 2.611 SECONDS TO 2.943 SECONDS.

4.2.6. BESSEL J0,J1,Y0,Y1, D.P.

4.2.6.1. IDENTIFICATION

BJY01/DOUBLE PRECISION BESSEL FUNCTIONS J0, J1, Y0, AND Y1

LANG	FILE	ELT/VERS	SIZE	ENTRY NAME
F-V	LIB*JPL$	BJY01/JPL	1010(8)=520(10)	BJY01

SUBROUTINES USED: DSQRT, DSIN, DCOS, DLOG, AND CHBPOL

COGNIZANT PERSON: E. W. NG, JPL, SECTION 314, 1969 AUGUST 1

4.2.6.2. PURPOSE

TO COMPUTE THE VALUES OF THE BESSEL FUNCTIONS J0, J1, Y0, AND Y1.

4.2.6.3. ACCURACY

FOR X .LE. N, AT LEAST 15 SIGNIFICANT DIGITS,

FOR X > N, AT LEAST 15 DECIMAL PLACES.

4.2.6.4. USAGE

```
      INTEGER                N0, N1

      DOUBLE PRECISION       X, B0(2), B1(2)

      CALL                   BJY01(X,N0,N1,B0,B1)
```

THE PARAMETERS ARE DEFINED AS FOLLOWS:

X ARGUMENT AT WHICH FUNCTION EVALUATION IS DESIRED. REQUIRE X > 0. IF X .LE. 0 THE SUBROUTINE WILL PRINT ON FORTRAN UNIT 6 THE MESSAGE:

FOR X=0., J0=1, J1=0., Y0=Y1=-INFINITY
AND WILL SET B0(1)=1., B1(1)=0., B0(2)=B1(2)=-1.D38.

N0 =0 DO NOT COMPUTE J0 OR Y0
=1 COMPUTE J0 BUT NOT Y0
=2 COMPUTE J0 AND Y0

N1 =0 DO NOT COMPUTE J1 OR Y1

```
        =1   COMPUTE  J1  BUT NOT  Y1
        =2   COMPUTE  J1  AND  Y1

B0(1)   COMPUTED VALUE OF  J0
B0(2)   COMPUTED VALUE OF  Y0
B1(1)   COMPUTED VALUE OF  J1
B1(2)   COMPUTED VALUE OF  Y1
```

4.2.6.5. REMARKS

THIS SUBROUTINE WAS ORIGINALLY DESIGNED FOR POSITIVE X. IF IT IS DESIRED TO INCLUDE ITS UTILITY FOR NEGATIVE X AS WELL, THE FOLLOWING IDENTITIES SHOULD BE USED:

```
J0(-X) = J0(X),     J1(-X) = -J1(X)
Y0(-X) = Y0(X) + 2*SQRT(-1)*J0(X)
Y1(-X) = -Y1(X) + 2*SQRT(-1)*J1(X)
```

NOTE THAT Y OF NEGATIVE ARGUMENT IS IN GENERAL COMPLEX.

(REFERENCE: NBS HANDBOOK OF MATH'L FUNCTIONS, APPLIED MATH SERIES NO. 55, 1964, P. 361)

4.2.9. BESSEL I AND K OF GENERAL ORDER, D.P.

4.2.9.1. IDENTIFICATION

DOUBLE PRECISION BESSEL FUNCTIONS I AND K OF GENERAL ORDER NU AND ARGUMENT X.

LANG	FILE	ELT/VERS	SIZE	ENTRY NAMES
F-V	LIB*JPLS	BESI/JPL	1774(8)=1020(10)	BESI, BESK

SUBROUTINES USED: DSIN, DLOG, DEXP

COGNIZANT PERSON: E. W. NG, JPL, SECTION 314, 1969 JULY 23

4.2.9.2. PURPOSE

TO COMPUTE IN DOUBLE PRECISION ARITHMETIC THE VALUES OF BESSEL FUNCTIONS I AND K OF THE GENERAL ORDER NU AND ARGUMENT X.

4.2.9.3. METHOD

RECURRENCE TECHNIQUES AND THE PHASE-AMPLITUDE METHOD ARE USED. (SEE REFERENCES)

4.2.9.4. ACCURACY

FOR NU .GE. X, A MINIMUM ACCURACY OF 15 SIGNIFICANT FIGURES IS FOUND; FOR NU < X, A MINIMUM ACCURACY OF 15 DECIMAL PLACES IS INSURED.

4.2.9.5. USAGE

```
      DOUBLE PRECISION  X, V, A(LDIMA), AK(LDIMA)
      INTEGER           N, LDIMA

      CALL  BESI(X,V,N,A,LDIMA)
 OR   CALL  BESK(X,V,N,A,AK,LDIMA)
```

THE ENTRY BESI COMPUTES THE FUNCTION DEXP(-X) * I OF ARGUMENT X AND OF ORDERS V, V+1,...,V+N.

THE ENTRY BESK COMPUTES BOTH THE FUNCTIONS DEXP(-X) * I AND DEXP(X) * K OF ARGUMENT X AND OF THE ORDERS V, V+1,...,V+N.

THE SUBROUTINE PARAMETERS ARE DEFINED AS FOLLOWS:

X THE ARGUMENT AT WHICH FUNCTION EVALUATION IS DESIRED; X .GE. 0.D0

V THE NON-INTEGER PART OF THE ORDER OF THE BESSEL FUNCTIONS; 0.D0 .LE. V .LE. 1.D0

N N+V IS THE HIGHEST ORDER OF THE BESSEL FUNCTIONS DESIRED; N MUST BE A POSITIVE INTEGER.

A(LDIMA) A(K), K=1,...,(N+1) ARE THE LOCATIONS IN WHICH THE VALUES FOR THE BESSEL FUNCTION DEXP(-X) * I OF ORDER V, (V+1),..., (V+N) ARE RETURNED. A( ) MUST HAVE DIMENSION .GE. LDIMA

AK(LDIMA) AK(K), K=1,..., (N+1) ARE THE LOCATIONS IN WHICH THE VALUES FOR THE BESSEL FUNCTION DEXP(X) * K OF THE ORDERS V, (V+1),..., (V+N) ARE RETURNED. AK( ) MUST HAVE DIMENSION .GE. LDIMA

LDIMA DIMENSION OF A AND AK; IT MUST BE AT LEAST MAX(IDINT(X),N) + 20 OR MAX(IDINT(2*X),N) FOR X > 20.

4.2.9.6. REMARKS

FOR X, V, OR N BEYOND ITS REQUIRED RANGE OR LDIMA NOT LARGE ENOUGH, THE MESSAGE 'ERROR IN BESSEL' AND THE VALUES OF X, V, N, NU, AND LDIMA ARE PRINTED. THE NUMBER (N+2) IS THE DIMENSION OF A NEEDED. IN THE CASE X OR V IS OUT OF RANGE, THE PARAMETER NU IS IRRELEVANT.

4.2.9.7. REFERENCES

M. GOLDSTEIN AND R.M. THALER, 'RECURRENCE TECHNIQUES FOR THE CALCULATION OF BESSEL FUNCTIONS', MTAC, VOL. XIII, PP. 102-108.

M. GOLDSTEIN AND R. M. THALER, 'BESSEL FUNCTIONS FOR LARGE ARGUMENTS', MTAC, VOL. XII, 1958, PP. 18-26.

D. JORDAN, ARGONNE NATIONAL LAB., PROGRAM WRITE-UP NO. C3715, NOV. 1967.

4.2.13. EXPONENTIAL INTEGRAL

4.2.13.1. IDENTIFICATION

EXPONENTIAL INTEGRAL

LANG	FILE	ELT/VERS	SIZE	ENTRY NAMES
F-V	LIB*JPL$	DEI/JPL		DEI

SUBROUTINES USED: DEXP,DLOG

COGNIZANT PERSON: E.W. NG, JPL, SECTION 314, 1970, SEPT. 14

4.2.13.2. PURPOSE

COMPUTE THE EXPONENTIAL INTEGRAL IN DOUBLE PRECISION ARITHMETIC. FOR X .GT. 0, THE EXPONENTIAL INTEGRAL, EI, IS DEFINED AS

EI(X) = INTEGRAL FROM T=-INFINITY TO T=X OF
(EXP(T)/T)*DT

WHERE THE INTEGRAL IS TO BE INTERPRETED AS THE CAUCHY PRINCIPAL VALUE. FOR X .LT. 0, EI(X) =-E1(-X), WHERE

E1(Z)=INTEGRAL FROM T=Z TO T=INFINITY OF
(EXP(-T)/T)*DT

4.2.13.3. METHOD

THIS SUBROUTINE COMPUTES THE EXPONENTIAL INTEGRAL BY CHEBYSHEV RATIONAL APPROXIMATIONS FROM W.J. CODY AND H.C. THACHER, JR., MATH. COMP. VOL. 22, PP. 641-650, AND VOL. 23, PP. 289-303.

4.2.13.4. ACCURACY

EXTENSIVE TESTS WERE PERFORMED ON THE UNIVAC 1108 AND THE FOLLOWING ACCURACY STATISTICS WERE FOUND:

INTERVAL OF X	MAXIMUM RELATIVE	RMS RELATIVE
(-150, -4)	2.20-16	5.1D-17

(-4, -1)	7.[illegible]D-17	1.2D-17
(-1, 0)	8.7D-18	1.4D-18
(0, 0.5)	1.8D-16*	3.2D-17*
(0.5, 6)	5.5D-17	1.0D-17
(6, 12)	1.6D-17	3.0D-18
(12, 24)	2.9D-17	7.1D-18
(24, 100)	8.9D-17	1.9D-17

CF. E.W. NG, COMM. ACM VOL. 13, #7, PP. 448-449.

4.2.13.5. USAGE

DOUBLE PRECISION DEI, X

USE THE FOLLOWING FUNCTION IN A FORTRAN ARITHMETIC STATEMENT:

DEI(X)

WHERE DEI IS THE COMPUTED VALUE OF THE EXPONENTIAL INTEGRAL.

4.2.13.6. LIMITATIONS

EI(0)=-INFINITY IS APPROXIMATED BY -7.2D75 AND EI(X .GT. 174.673) IS APPROXIMATED BY +7.2D75.

THESE LIMITATIONS WERE ORIGINALLY IMPOSED FOR THE IBM/360 AND HAVE NOT BEEN MODIFIED FOR THE UNIVAC 1108.

4.2.13.7. ERROR EXITS

NONE

PPLOT

Reference:

D. P. Wei, "A Printer Plot Package for the Univac U1106", Research Memo 72-1K4-COMPS-M16 (9-8-72).

Purpose:

This package of subroutines is used to generate graphs on the line printer. It produces an 8 by 10 grid plot; a grid is 6 print characters high and 10 characters wide. Data points are plotted as a symbol (e.g., *, A, 3) which is specified by the user. One or more plots can be included in a single graph.

Usage:

The package consists of three Fortran V subroutines.

1. PSCALE scans data for the minimum and maximum values, and establishes a range of values which produces a rational scale for the axis. The scale is chosen such that each grid is 1, 2, or 3 times an appropriate power of 10.

2. PSETUP sets up the work area for the plot. The scales for the abscissa and ordinate, and the labelling information for the axes are stored. PSETUP must be called to initialize each graph.

3. PPLOT stores plot data and generates the plot. An option controls initiation of plotting. Thus, multiple calls on PPLOT can be made to store data for several plots before a request for plotting is made.

The parameter specifications for each subroutine follow.

PSCALE

Subroutine Specification:

```
    SUBROUTINE PSCALE (V, NPTS, VLOW, VHIGH, FIRST, IXORY)
        DIMENSION V(1)
        INTEGER NPTS, IXORY
        LOGICAL FIRST
        REAL VLOW, VHIGH
```

Purpose:

To scan the data in vector V, and obtain the minimum and maximum values. These values are then used to establish a range which produces a rational scale for the plot axis.

Usage:

The Fortran calling sequence is:

```
    CALL PSCALE(V, NPTS, VLOW, VHIGH, FIRST, IXORY)
```

where:

- V = REAL vector containing data to be scanned.
- NPTS = number of points in V to be scanned.
- VLOW,VHIGH = output variables into which the routine returns the lower and upper values for the range of V.
- FIRST = LOGICAL variable or value indicating when it is the first set of values to be scanned. The value is true (.TRUE.) if there is a single vector to be scanned. For multiple vectors (when more than one plot is to be included in a graph), a call on PSCALE must be made for each plot (vector). The first call will have the value .TRUE.; for subsequent calls, the value will be .FALSE. The range defined by the final values of VLOW and VHIGH will include all the values scanned.
- IXORY = indicator whether abscissa range (x) or ordinate range (y) is to be established. Specify the value 0 for abscissa and 1 for ordinate.

Note:

The abscissa axis is 10 grid wide. The ordinate axis is 8 grid high. PSCALE need not be used if the range of values for the abscissa (or ordinate) is known. The number of grid positions should be taken into consideration for proper annotation of the axis.

PSETUP

Subroutine Specification:

```
    SUBROUTINE (XLOW, XHIGH, YLOW, YHIGH, IEOF, IGRID, LBLX,
               LBLY)
        REAL XLOW, XHIGH, YLOW, YHIGH
        INTEGER IEOF, IGRID
        DIMENSION LBLX(1), LBLY(1)
```

Purpose:

To setup work area before data values to be plotted are stored (via PPLOT routine). Labelling information, ranges of values for the abscissa and ordinate variables, and options for grid and annotation are supplied as input parameters.

Usage:

The Fortran calling sequence is:

```
CALL PSETUP (XLOW, XHIGH, YLOW, YHIGH, IEOF, IGRID,
LBLX, LBLY)
```

where:

XLOW,XHIGH = the lower and upper values for the range of the abscissa variable (x).

YLOW,YHIGH = the lower and upper values for the range of the ordinate variable (y).

IEOF = Hollerith option to set either E or F format for annotating the axes. Specify the Hollerith constant 1HE for E-format (1PE10.3), or 1HF for F-format (F10.3).

IGRID = Hollerith option to select grid. Specify the Hollerith constant 1HG for grid; otherwise, specify 1Hb (b denotes a blank or space).

LBLX = INTEGER vector containing information for labelling the x-axis. The first element LBLX(1) contains the number of characters in the label. The labelling information is stored, six characters per word, starting in the second element LBLX(2). The maximum label length is 30 characters.

LBLY = INTEGER vector containing information for labelling the y-axis. The rules for storing the information is the same as described for LBLX. The maximum label length is 24 characters.

Restrictions:

1. The value of XHIGH must be equal to or greater than XLOW.

2. The value of YHIGH must be equal to or greater than YLOW.

3. Labelling information will be truncated if the maximum label length is exceeded; i.e., 30 characters for x and 24 for y.

PPLOT

Subroutine Specification:

```
    SUBROUTINE PPLOT (IPLOT, ICHR, NPTS, XV, YV)
        LOGICAL IPLOT
        INTEGER ICHR, NPTS
        DIMENSION XV(1), YV(1)
```

Purpose:

To store plot data and initiate plotting. The IPLOT parameter controls plot initiation.

Usage:

The Fortran calling sequence is:

```
    CALL PPLOT (IPLOT, ICHR, NPTS, XV, YV)
```

where:

IPLOT = LOGICAL control variable for plotting. Specify .FALSE. if data values are to be stored and plotting is to be deferred; (used for multiple plots per graph). Specify .TRUE. if data values are to be stored and plotted.

ICHR = a single Hollerith character to be used as the plot symbol; (e.g., 1H*).

NPTS = number of data points to be plotted.

XV,YV = vectors containing coordinate values (x_i, y_i) to be plotted.

Restrictions:

1. If more than one data point occupies the same plot position, the most recent value processed will be used.

2. Data values outside the range specified by PSETUP will not be plotted. They will be listed below the plot.

SECOND

<u>Subroutine Specification</u>:

```
     SUBROUTINE SECOND(ISEC)
          INTEGER ISEC
```

<u>Programmed by</u>: L. C. Lintner

<u>Purpose</u>:

This routine provides the user with a means of determining the amount of central processor (CPU) time that he has used. This time is accumulated from the beginning of the run; i.e., it includes the CPU time accrued for all preceding tasks in the run setup.

<u>Usage</u>:

SECOND is an assembly language routine in "FORTRAN callable" form.

The FORTRAN calling sequence is:

```
    CALL SECOND(ISEC)
```

The parameter ISEC must be an integer variable. The result returned in ISEC is the integer number of CPU seconds used since the start of the run.

This routine may also be called as an integer function:

```
    L = NSEC(ISEC)
```

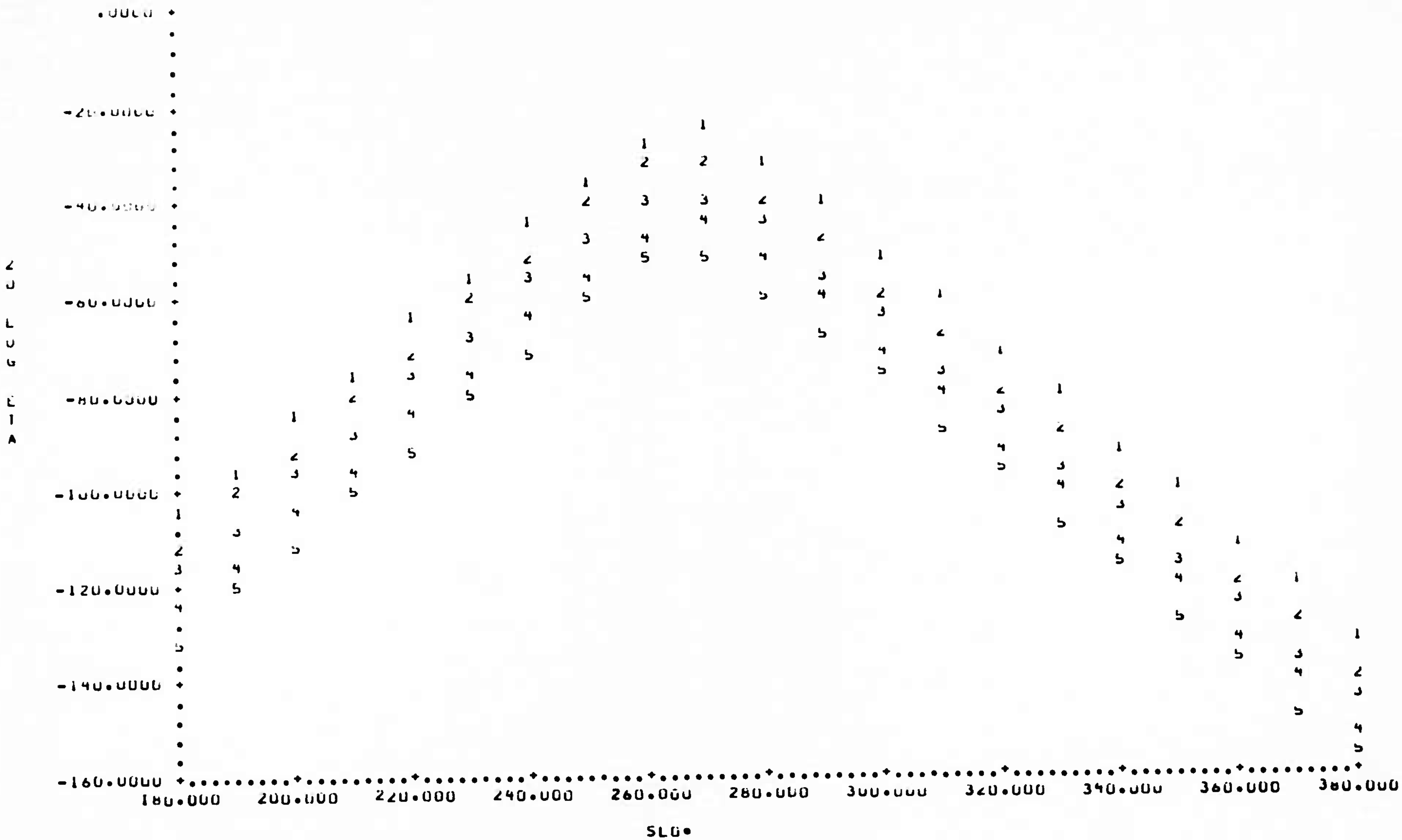
In this case, the integer number of seconds will be returned as the value of the function in addition to being stored in ISEC.

(This page intentionally left blank.)

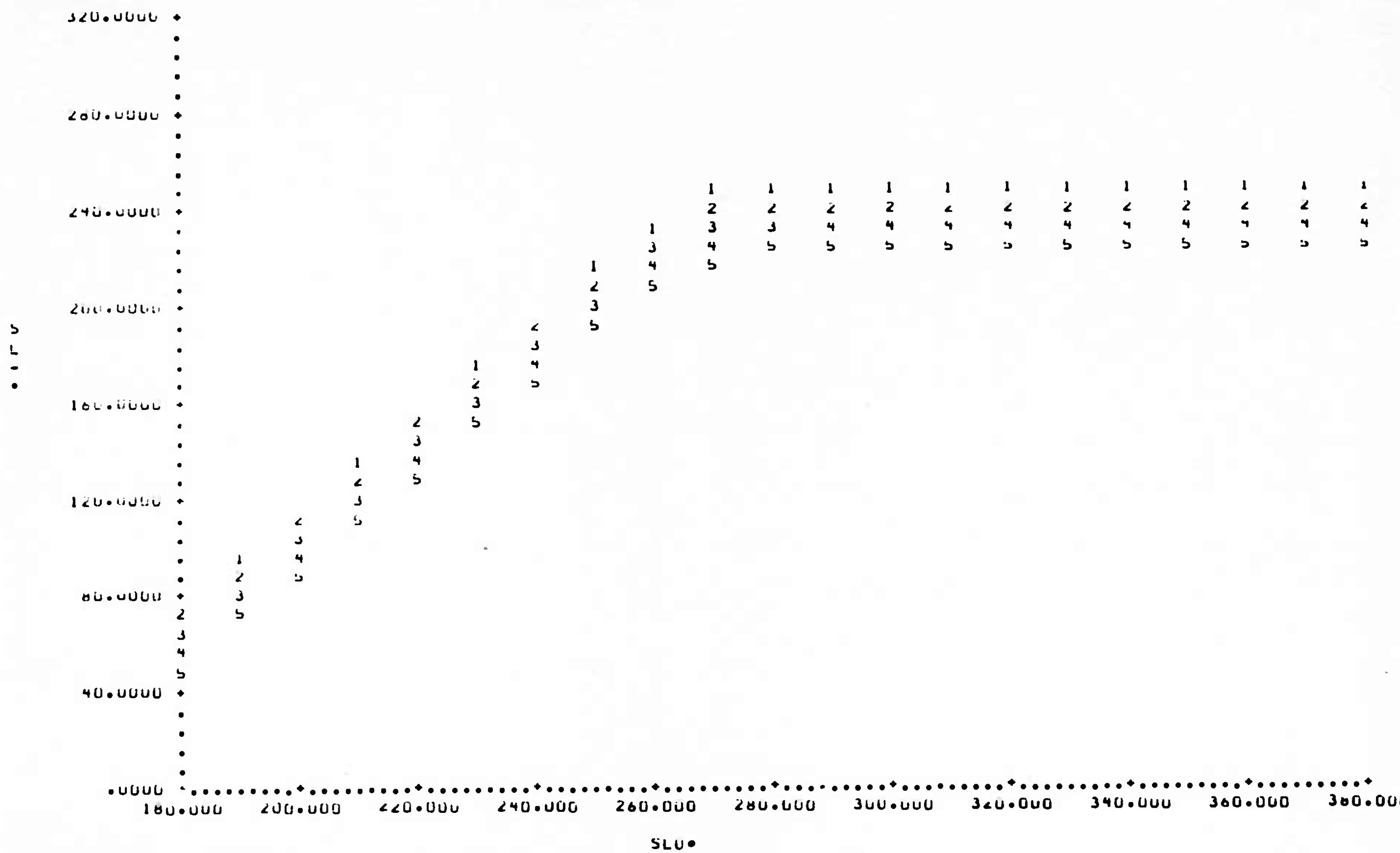
FIGURE 1

(Pages 37 to 85)

```
@XQT SONIC*NONLN.BEAM
 NOC=  8

     IOP          IOE          IOS          IOD          IOO
      0            1            1            0            1
    ALTRG        PIPZ
   .101-04      .100+01
  NFM=  5     NFMRD=  0


  FO/F- =    .500+01     EN=    .108+01
  FO/F- =    .100+02     EN=    .108+01
  FO/F- =    .200+02     EN=    .109+01
  FO/F- =    .400+02     EN=    .109+01
  FO/F- =    .800+02     EN=    .109+01
```

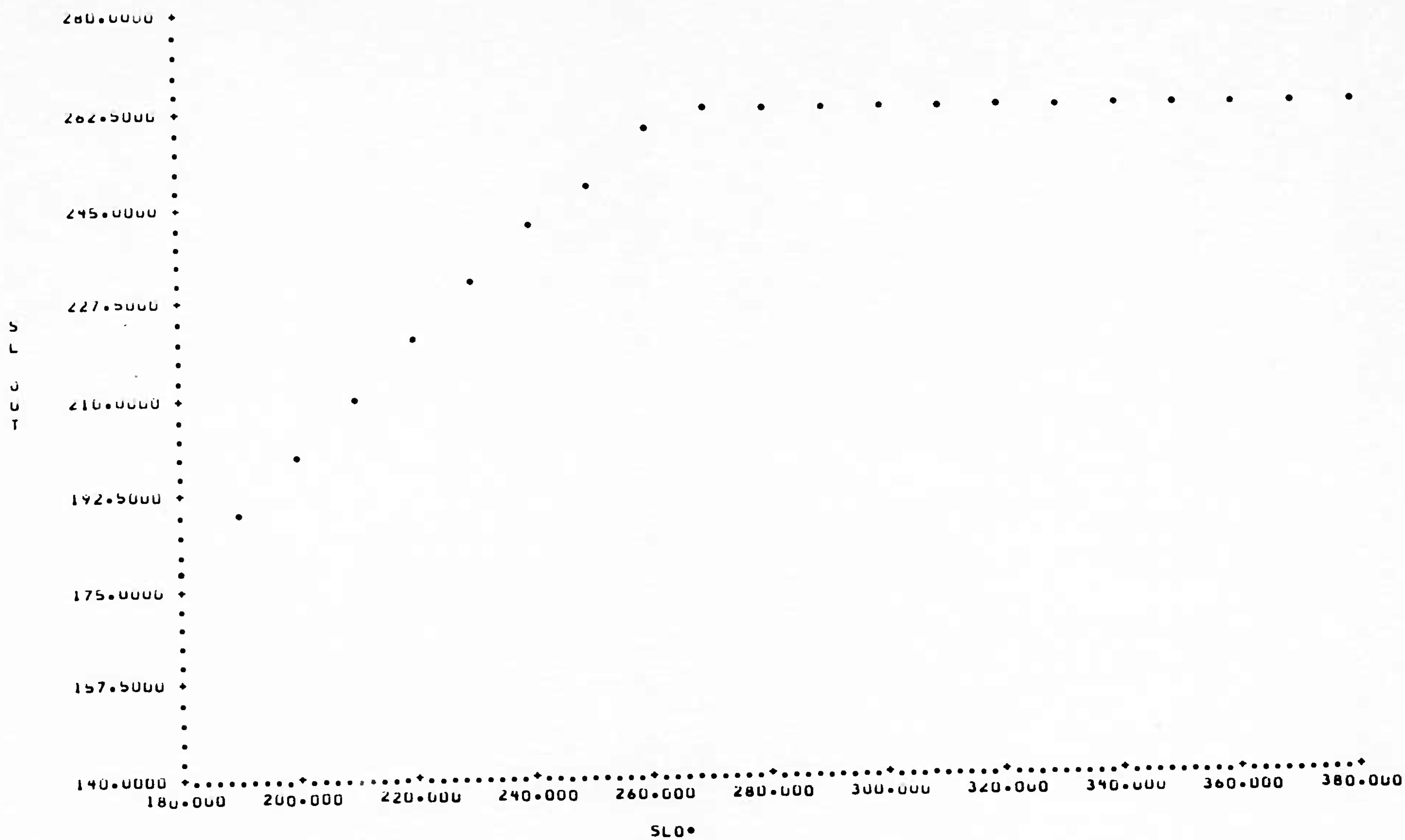


ALPHA(T)*RO= .100-04 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS



ALPHA(T)•RO= .100-04 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

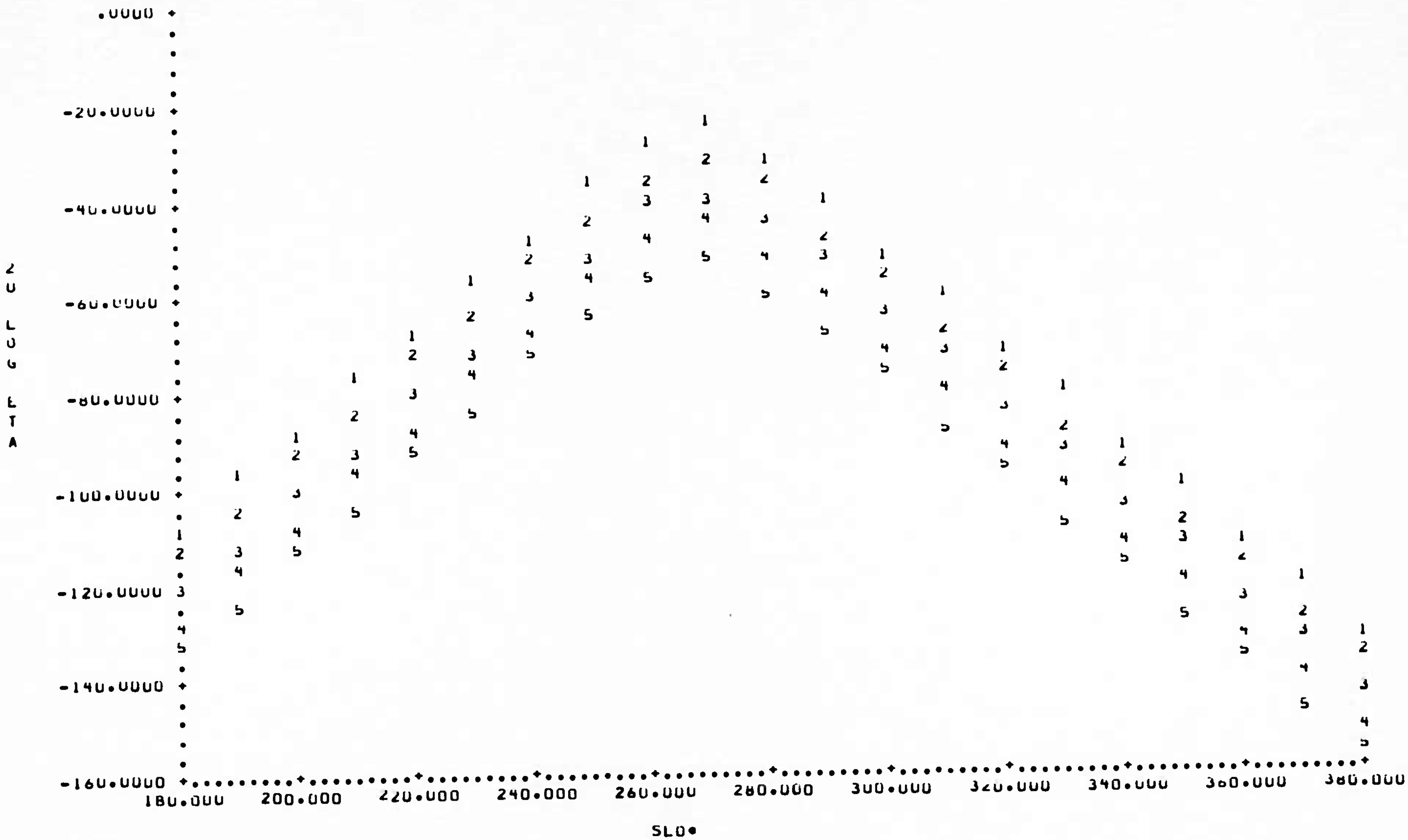
SL0.	SL-(1).	SL-(2).	SL-(3).	SL-(4).	SL-(5).
.180+03	.752+02	.686+02	.620+02	.554+02	.487+02
.190+03	.952+02	.886+02	.820+02	.754+02	.687+02
.200+03	.115+03	.109+03	.102+03	.954+02	.887+02
.210+03	.135+03	.129+03	.122+03	.115+03	.109+03
.220+03	.155+03	.149+03	.142+03	.135+03	.129+03
.230+03	.175+03	.169+03	.162+03	.155+03	.149+03
.240+03	.195+03	.189+03	.182+03	.175+03	.169+03
.250+03	.215+03	.208+03	.202+03	.195+03	.189+03
.260+03	.233+03	.227+03	.220+03	.214+03	.207+03
.270+03	.245+03	.238+03	.232+03	.225+03	.218+03
.280+03	.248+03	.241+03	.234+03	.228+03	.221+03
.290+03	.248+03	.242+03	.235+03	.228+03	.222+03
.300+03	.249+03	.242+03	.235+03	.229+03	.222+03
.310+03	.249+03	.242+03	.235+03	.229+03	.222+03
.320+03	.249+03	.242+03	.235+03	.229+03	.222+03
.330+03	.249+03	.242+03	.235+03	.229+03	.222+03
.340+03	.249+03	.242+03	.235+03	.229+03	.222+03
.350+03	.249+03	.242+03	.235+03	.229+03	.222+03
.360+03	.249+03	.242+03	.235+03	.229+03	.222+03
.370+03	.249+03	.242+03	.235+03	.229+03	.222+03
.380+03	.249+03	.242+03	.235+03	.229+03	.222+03



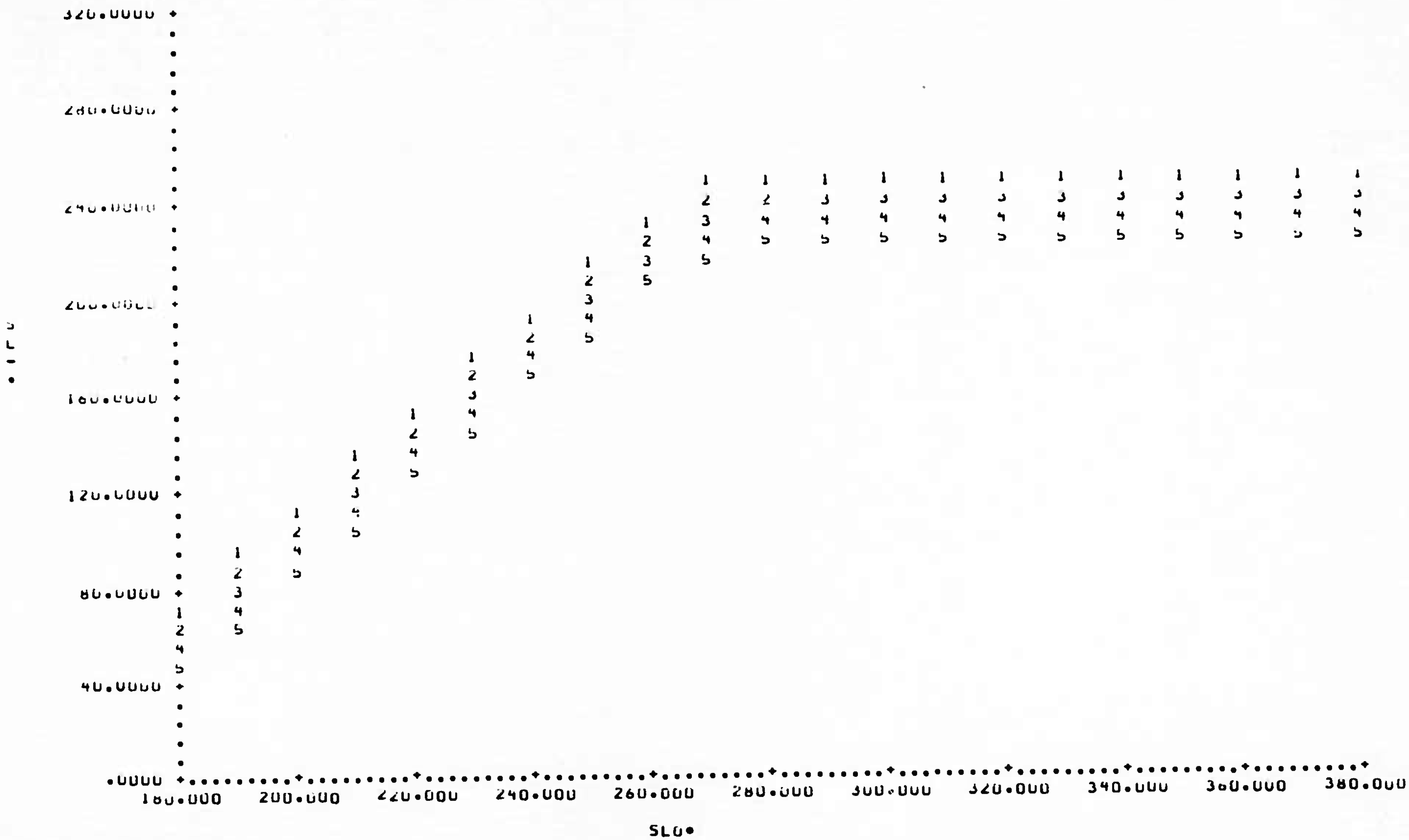
ALPHA(U)•RO= .577-06 NEPERS
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

SLO.	SL OUT
.180+03	.180+03
.190+03	.190+03
.200+03	.200+03
.210+03	.210+03
.220+03	.220+03
.230+03	.230+03
.240+03	.240+03
.250+03	.250+03
.260+03	.258+03
.270+03	.262+03
.280+03	.263+03
.290+03	.263+03
.300+03	.263+03
.310+03	.263+03
.320+03	.263+03
.330+03	.263+03
.340+03	.263+03
.350+03	.263+03
.360+03	.263+03
.370+03	.263+03
.380+03	.263+03

```
   IOP          IOE          IOS          IOD          IOO
    0            1            1            0            1
  ALTRO        PIP2
 .100-03      .100+01
NFR=   5    NFMRD=   0


FO/F- =    .500+01     EN=    .110+01
FO/F- =    .100+02     EN=    .110+01
FO/F- =    .200+02     EN=    .111+01
FO/F- =    .400+02     EN=    .111+01
FO/F- =    .800+02     EN=    .112+01
```

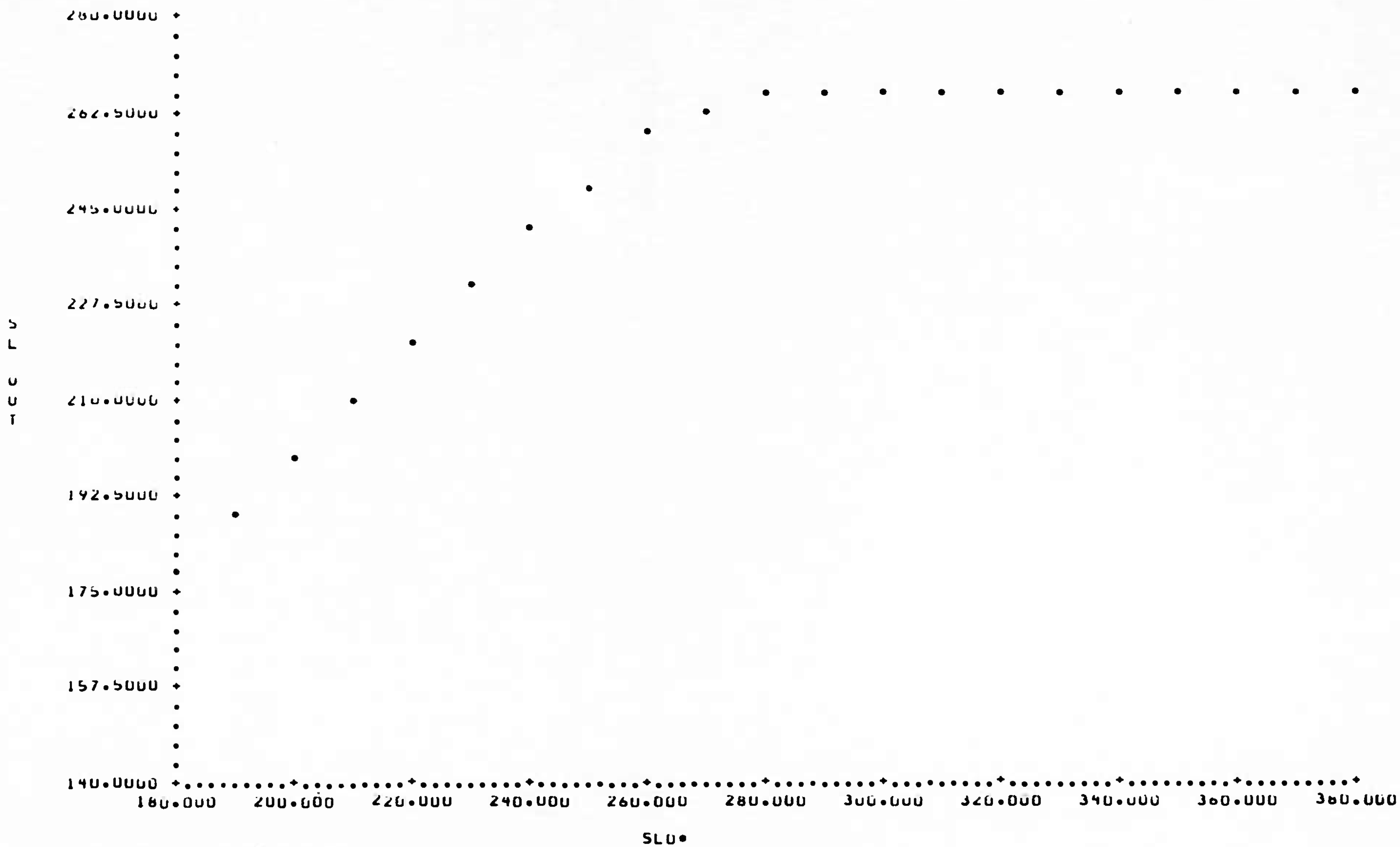


ALPHA(T)•RO= .100-03 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS



ALPHA(T)•RO= .100-03 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

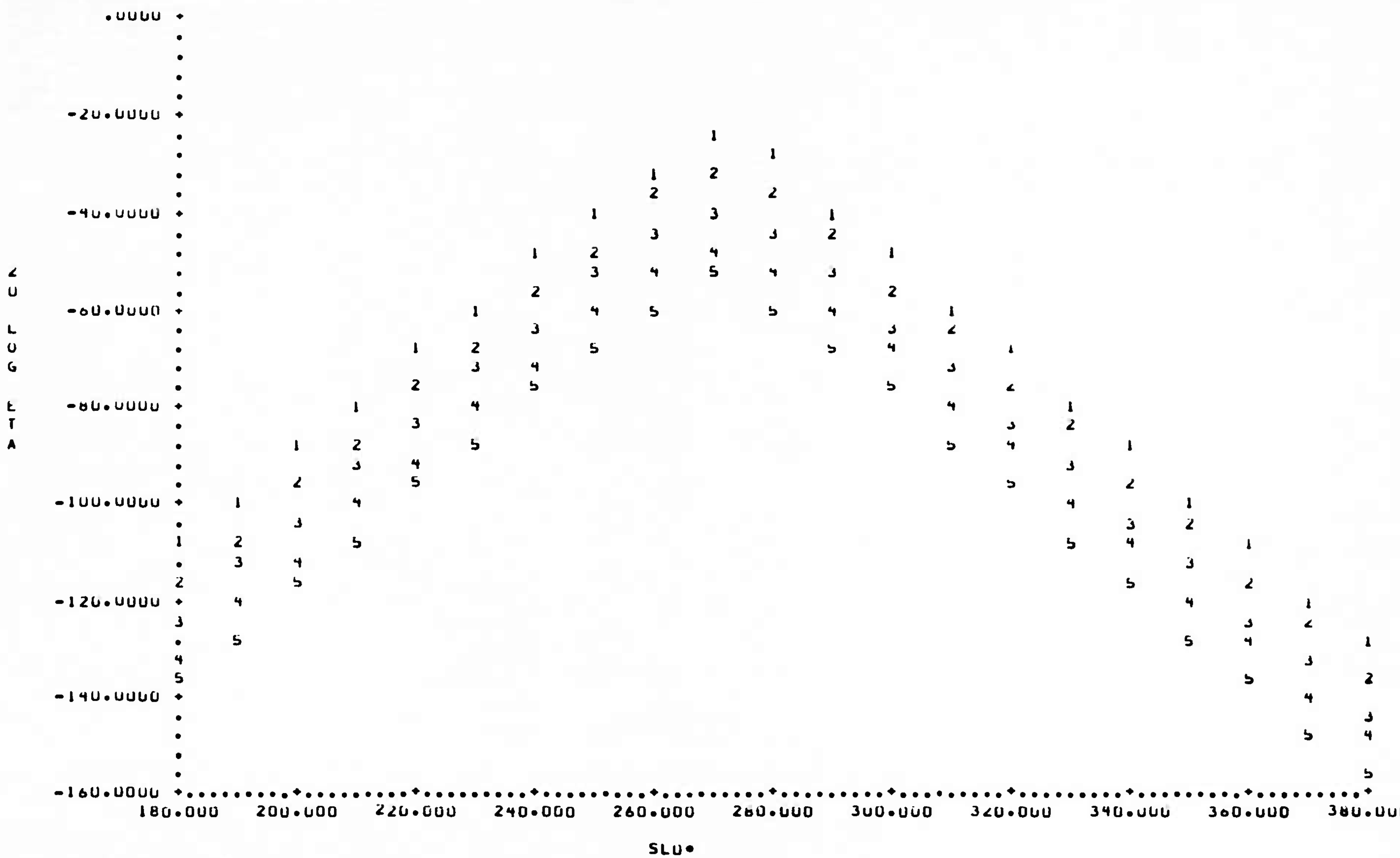
SL0.	SL-(1).	SL-(2).	SL-(3).	SL-(4).	SL-(5).
.180+03	.732+02	.665+02	.598+02	.530+02	.460+02
.190+03	.932+02	.865+02	.798+02	.730+02	.660+02
.200+03	.113+03	.107+03	.998+02	.930+02	.860+02
.210+03	.133+03	.127+03	.120+03	.113+03	.106+03
.220+03	.153+03	.147+03	.140+03	.133+03	.126+03
.230+03	.173+03	.167+03	.160+03	.153+03	.146+03
.240+03	.193+03	.187+03	.180+03	.173+03	.166+03
.250+03	.213+03	.206+03	.200+03	.193+03	.186+03
.260+03	.232+03	.225+03	.219+03	.212+03	.205+03
.270+03	.245+03	.238+03	.232+03	.225+03	.218+03
.280+03	.249+03	.242+03	.235+03	.228+03	.222+03
.290+03	.250+03	.243+03	.236+03	.229+03	.222+03
.300+03	.250+03	.243+03	.237+03	.230+03	.223+03
.310+03	.250+03	.243+03	.237+03	.230+03	.223+03
.320+03	.250+03	.243+03	.237+03	.230+03	.223+03
.330+03	.250+03	.243+03	.237+03	.230+03	.223+03
.340+03	.250+03	.243+03	.237+03	.230+03	.223+03
.350+03	.250+03	.243+03	.237+03	.230+03	.223+03
.360+03	.250+03	.243+03	.237+03	.230+03	.223+03
.370+03	.250+03	.243+03	.237+03	.230+03	.223+03
.380+03	.250+03	.243+03	.237+03	.230+03	.223+03



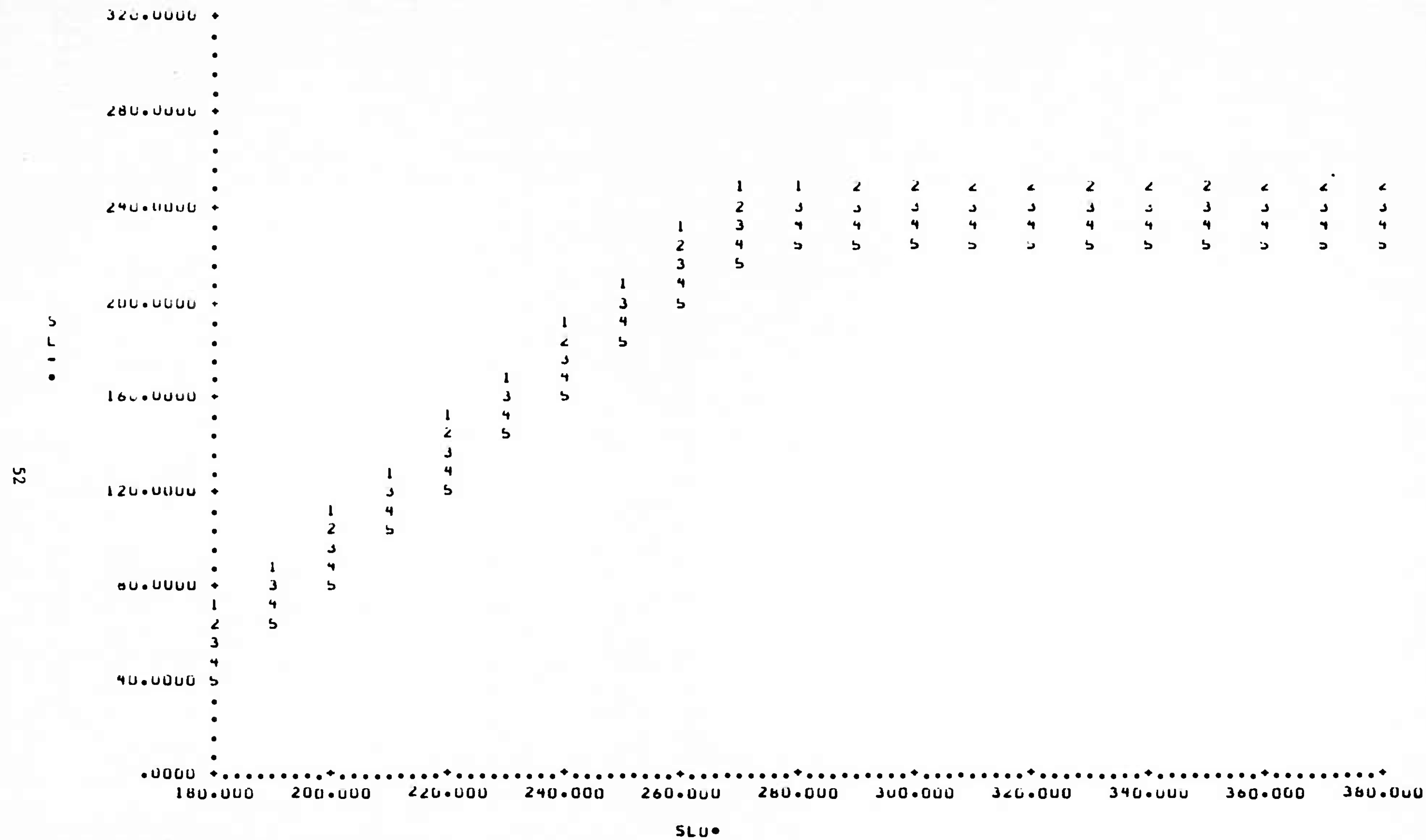
ALPHA(U)•RU= .577-05 NEPERS
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

SLO.	SL OUT
.180+03	.180+03
.190+03	.190+03
.200+03	.200+03
.210+03	.210+03
.220+03	.220+03
.230+03	.230+03
.240+03	.240+03
.250+03	.250+03
.260+03	.259+03
.270+03	.263+03
.280+03	.264+03
.290+03	.265+03
.300+03	.265+03
.310+03	.265+03
.320+03	.265+03
.330+03	.265+03
.340+03	.265+03
.350+03	.265+03
.360+03	.265+03
.370+03	.265+03
.380+03	.265+03

```
   10P        10C        105        100        100
    0          1          1          0          1
   ALTRO      PIP2
  .100-02    .100+01
NFM=  5    NFMRD=  0


FU/F- =   .500+01   EN=   .113+01
FU/F- =   .100+02   EN=   .114+01
FU/F- =   .200+02   EN=   .114+01
FU/F- =   .400+02   EN=   .115+01
FU/F- =   .800+02   EN=   .116+01
```

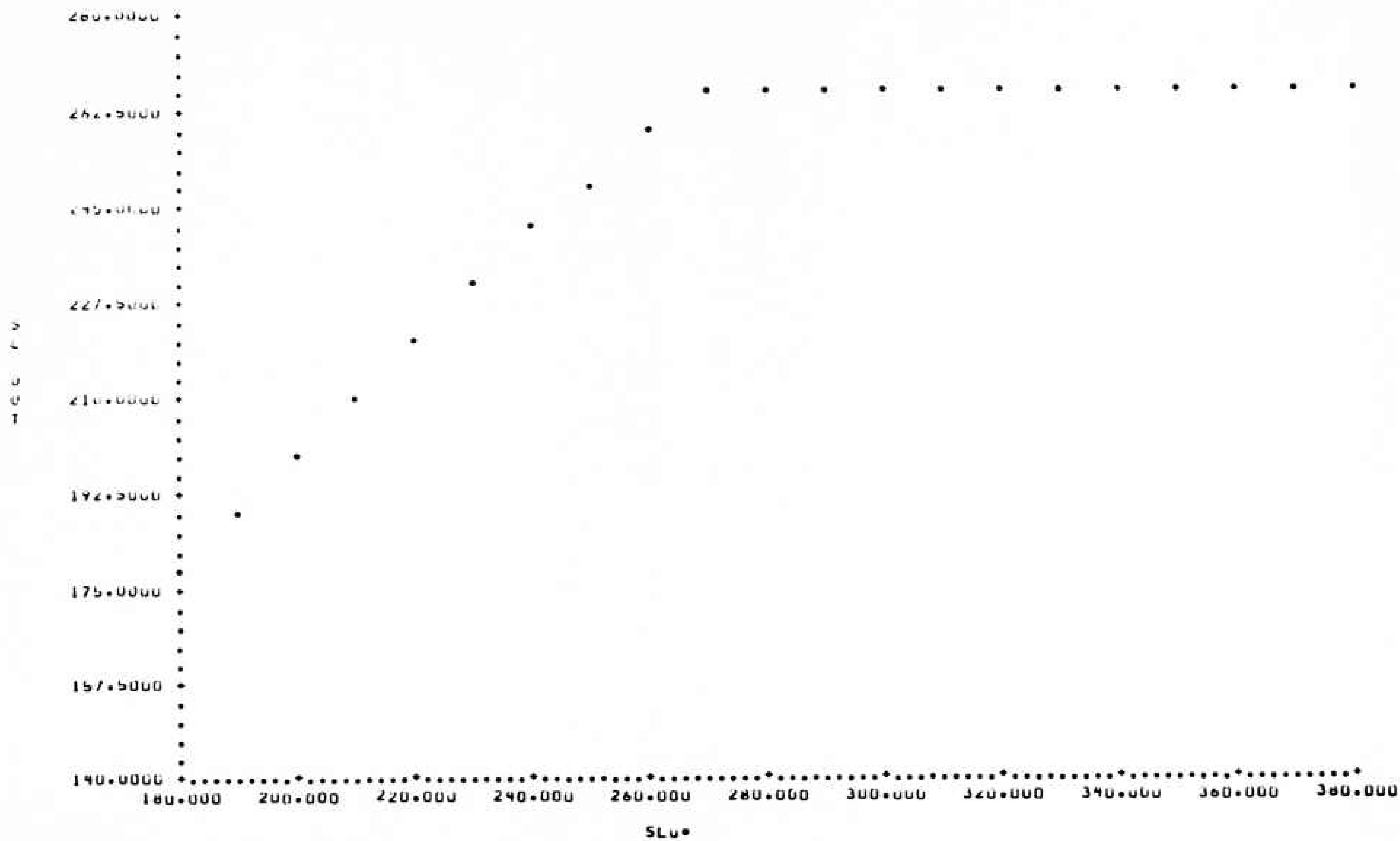


ALPHA(T)*RO= .100-02 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS



ALPHA(T)*RU= .100-02 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

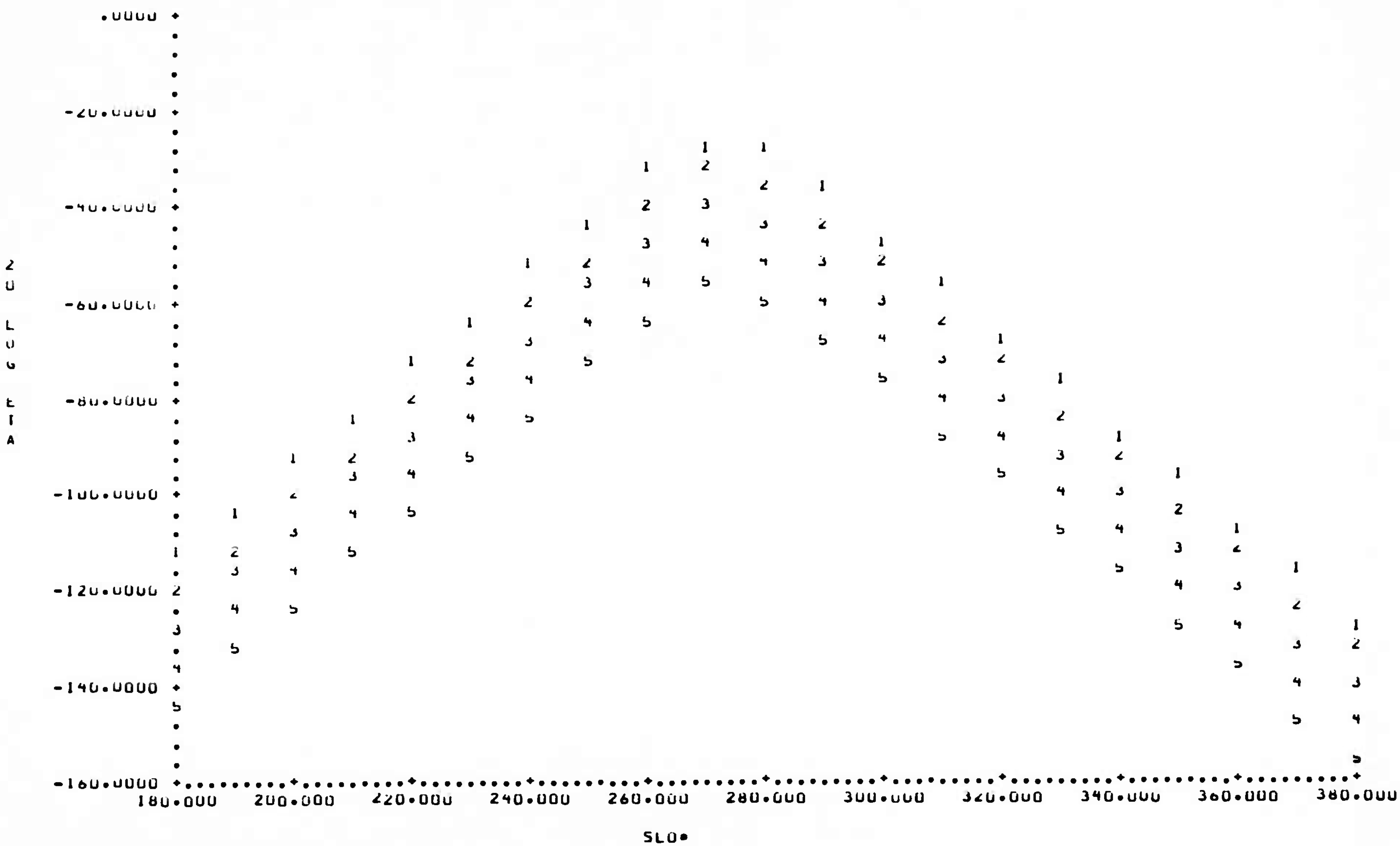
SLO.	SL-(1).	SL-(2).	SL-(3).	SL-(4).	SL-(5).
.180+03	.707+02	.638+02	.568+02	.496+02	.423+02
.190+03	.907+02	.838+02	.768+02	.696+02	.623+02
.200+03	.111+03	.104+03	.968+02	.896+02	.823+02
.210+03	.131+03	.124+03	.117+03	.110+03	.102+03
.220+03	.151+03	.144+03	.137+03	.130+03	.122+03
.230+03	.171+03	.164+03	.157+03	.150+03	.142+03
.240+03	.191+03	.184+03	.177+03	.170+03	.162+03
.250+03	.211+03	.204+03	.197+03	.190+03	.182+03
.260+03	.230+03	.223+03	.216+03	.209+03	.202+03
.270+03	.245+03	.236+03	.231+03	.224+03	.216+03
.280+03	.250+03	.243+03	.236+03	.229+03	.222+03
.290+03	.251+03	.244+03	.237+03	.230+03	.223+03
.300+03	.252+03	.245+03	.238+03	.230+03	.223+03
.310+03	.252+03	.245+03	.238+03	.230+03	.223+03
.320+03	.252+03	.245+03	.238+03	.230+03	.223+03
.330+03	.252+03	.245+03	.238+03	.230+03	.223+03
.340+03	.252+03	.245+03	.238+03	.230+03	.223+03
.350+03	.252+03	.245+03	.238+03	.230+03	.223+03
.360+03	.252+03	.245+03	.238+03	.230+03	.223+03
.370+03	.252+03	.245+03	.238+03	.230+03	.223+03
.380+03	.252+03	.245+03	.238+03	.230+03	.223+03



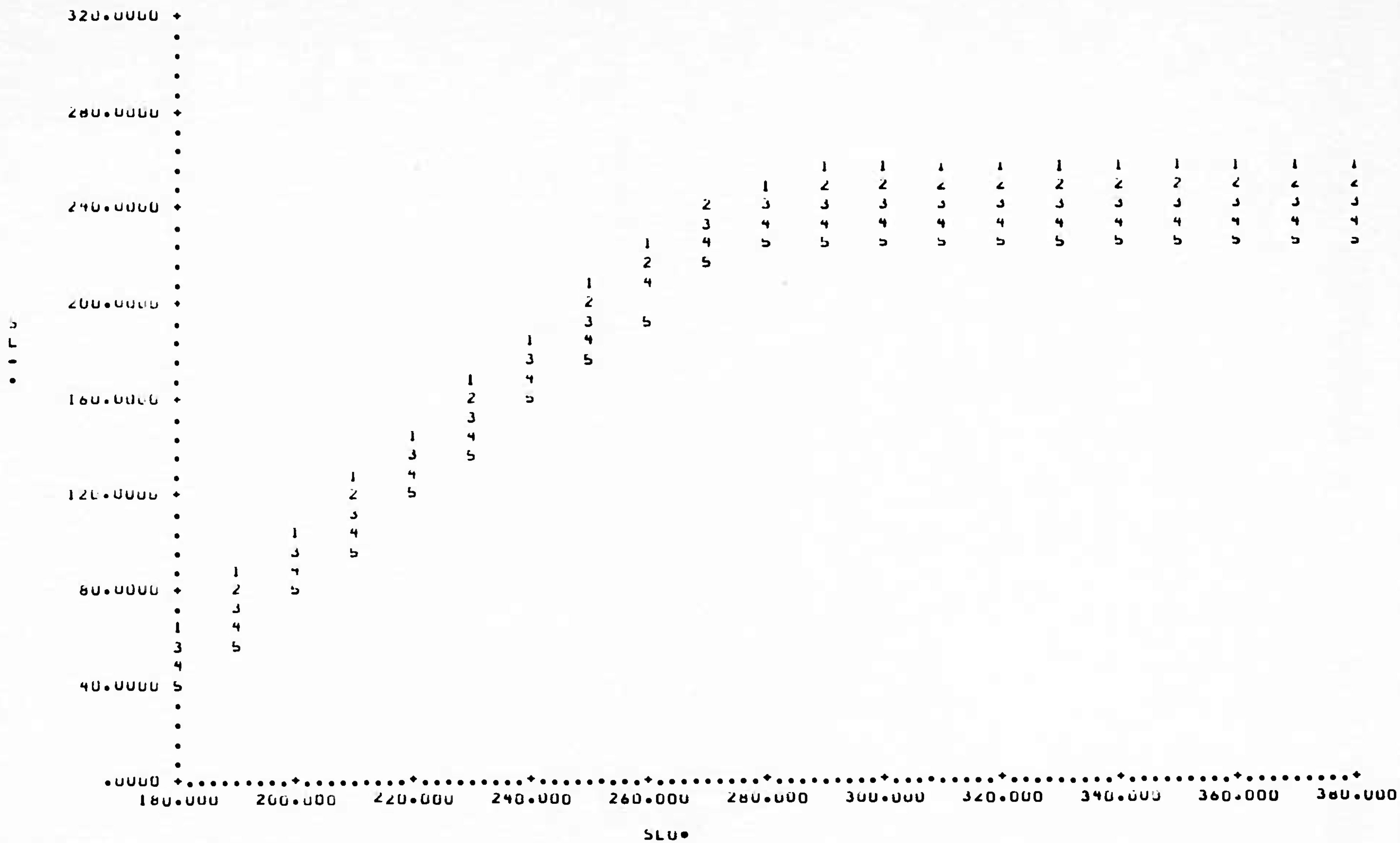
ALPHA(U)*RO=   .577-04 NEPERS
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

SLO.	SL OUT
.180+03	.180+03
.190+03	.190+03
.200+03	.200+03
.210+03	.210+03
.220+03	.220+03
.230+03	.230+03
.240+03	.240+03
.250+03	.250+03
.260+03	.259+03
.270+03	.265+03
.280+03	.266+03
.290+03	.267+03
.300+03	.267+03
.310+03	.267+03
.320+03	.267+03
.330+03	.267+03
.340+03	.267+03
.350+03	.267+03
.360+03	.267+03
.370+03	.267+03
.380+03	.267+03

```
   IOP        IOE        IOS        IOO        IOO
    0          1          1          0          1
  ALTRD       PIP2
 .100-01     .100+01
NFM=  5    NFMRD=  0


FO/F- =   .500+01    EN=   .118+01
FO/F- =   .100+02    EN=   .120+01
FO/F- =   .200+02    EN=   .121+01
FO/F- =   .400+02    EN=   .123+01
FO/F- =   .800+02    EN=   .125+01
```

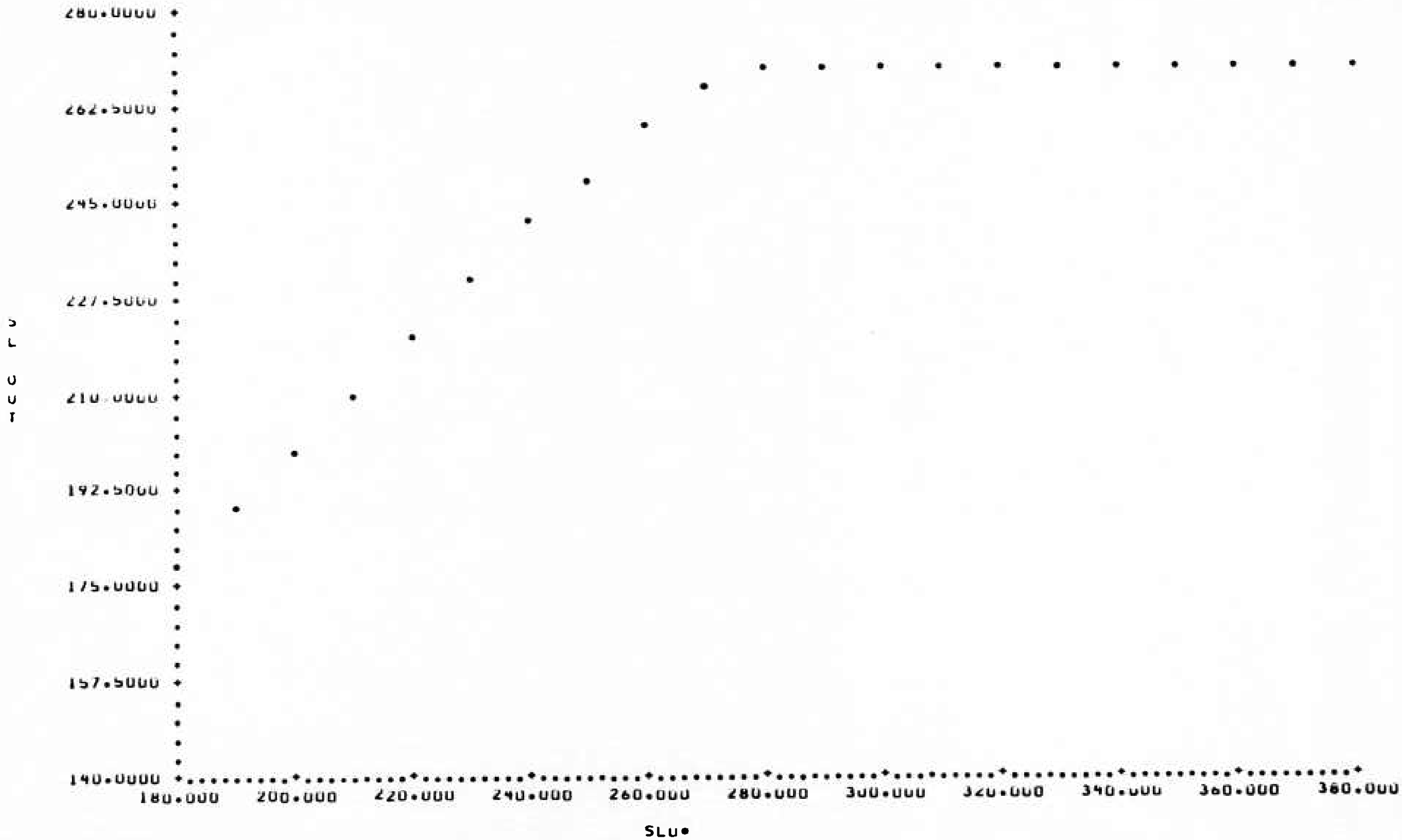


ALPHA(T)*RO= .100-01 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS



ALPHA(T)•RU= .100-01 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PA: AL (KHZ) RMS

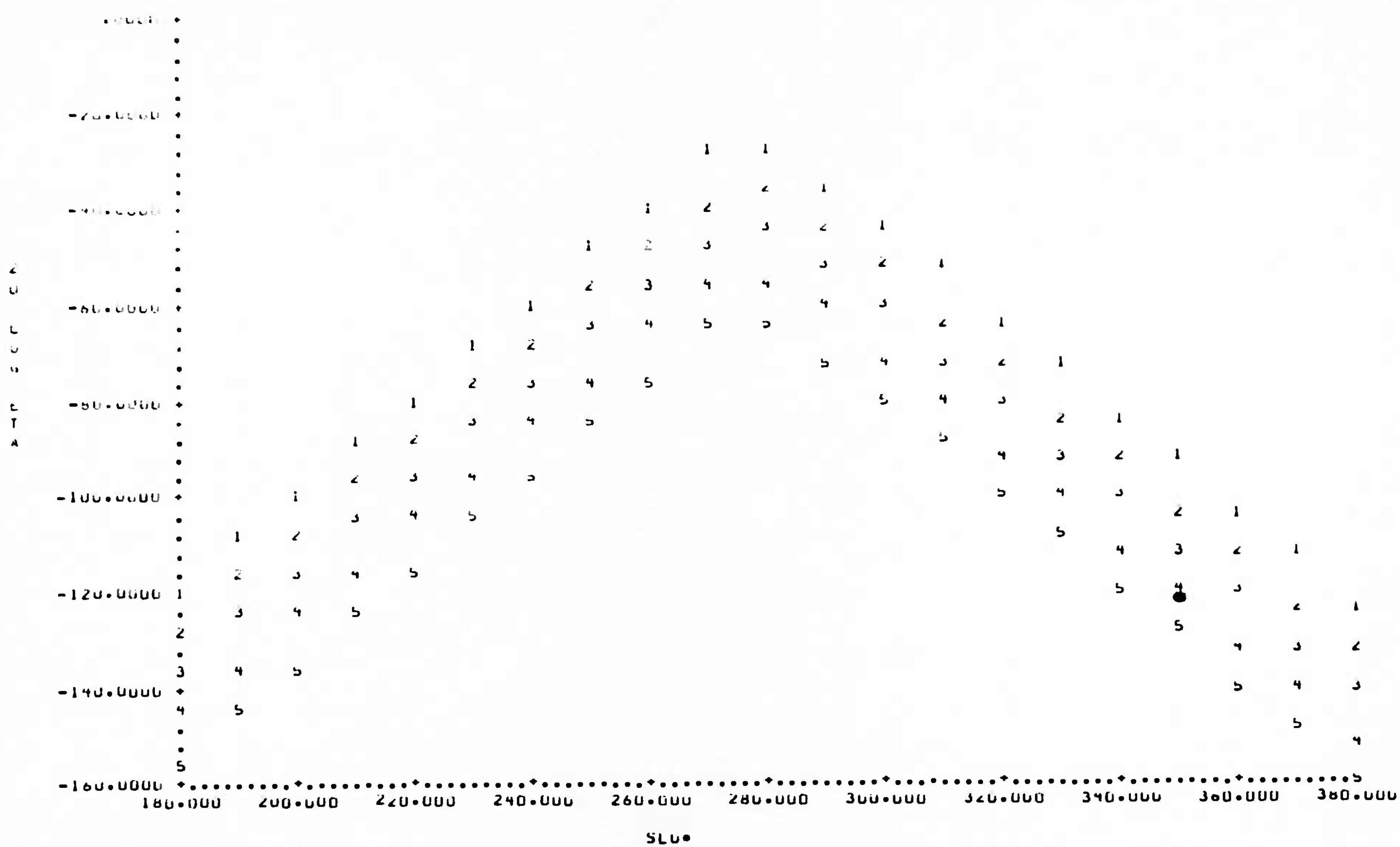
SLD.	SL-(1).	SL-(2).	SL-(3).	SL-(4).	SL-(5).
.180+03	.673+02	.599+02	.523+02	.444+02	.363+02
.190+03	.873+02	.799+02	.723+02	.644+02	.563+02
.200+03	.107+03	.999+02	.923+02	.844+02	.763+02
.210+03	.127+03	.120+03	.112+03	.104+03	.963+02
.220+03	.147+03	.140+03	.132+03	.124+03	.116+03
.230+03	.167+03	.160+03	.152+03	.144+03	.136+03
.240+03	.187+03	.180+03	.172+03	.164+03	.156+03
.250+03	.207+03	.200+03	.192+03	.184+03	.176+03
.260+03	.227+03	.219+03	.212+03	.204+03	.196+03
.270+03	.244+03	.236+03	.229+03	.221+03	.213+03
.280+03	.251+03	.244+03	.236+03	.229+03	.220+03
.290+03	.253+03	.246+03	.238+03	.230+03	.222+03
.300+03	.254+03	.246+03	.239+03	.231+03	.223+03
.310+03	.254+03	.246+03	.239+03	.231+03	.223+03
.320+03	.254+03	.246+03	.239+03	.231+03	.223+03
.330+03	.254+03	.246+03	.239+03	.231+03	.223+03
.340+03	.254+03	.246+03	.239+03	.231+03	.223+03
.350+03	.254+03	.246+03	.239+03	.231+03	.223+03
.360+03	.254+03	.246+03	.239+03	.231+03	.223+03
.370+03	.254+03	.246+03	.239+03	.231+03	.223+03
.380+03	.254+03	.246+03	.239+03	.231+03	.223+03



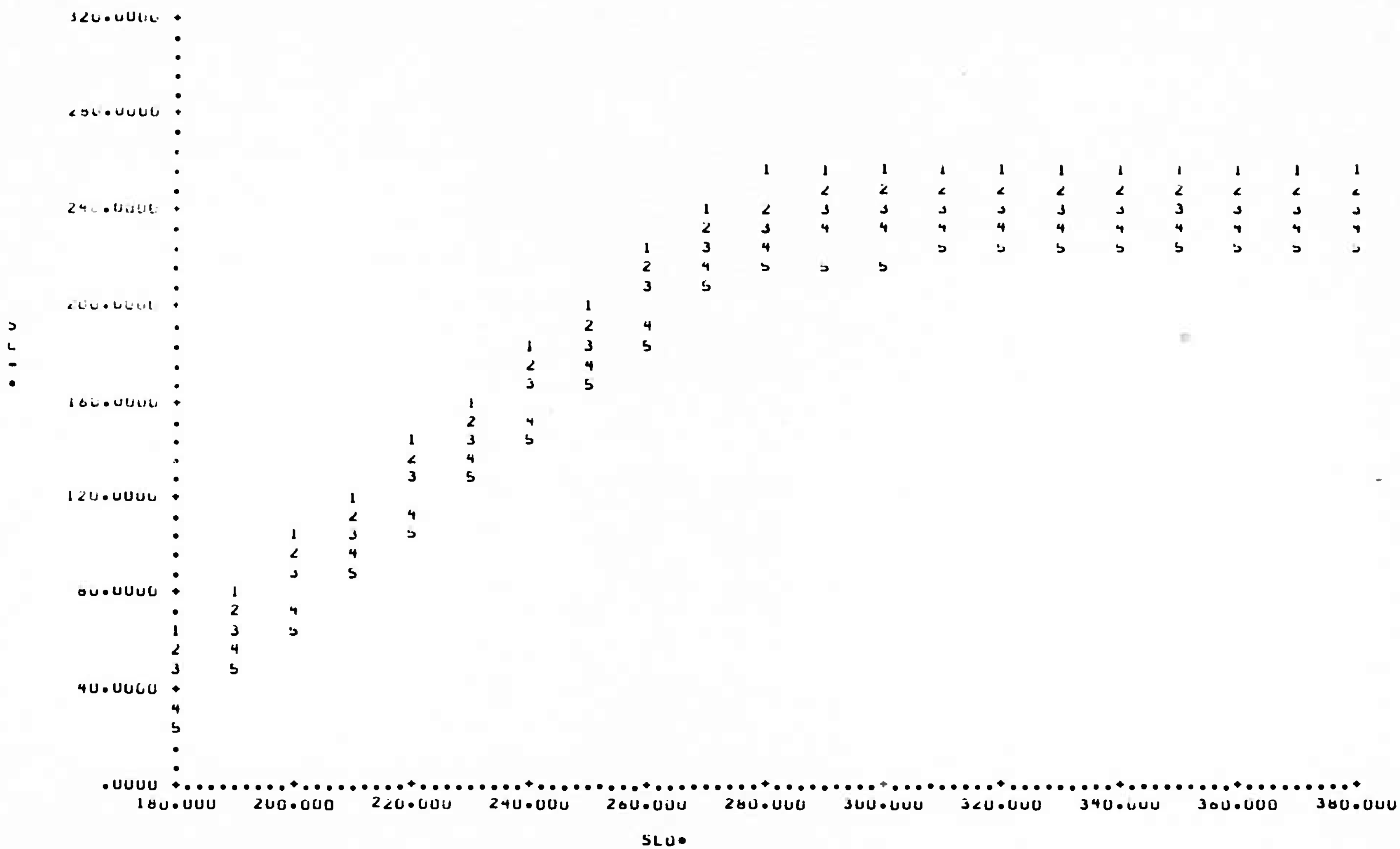
ALPHA(U)*RO= .577-03 NEPERS
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

SLO.	SL OUT
.180+03	.180+03
.190+03	.190+03
.200+03	.200+03
.210+03	.210+03
.220+03	.220+03
.230+03	.230+03
.240+03	.240+03
.250+03	.250+03
.260+03	.259+03
.270+03	.266+03
.280+03	.269+03
.290+03	.259+03
.300+03	.269+03
.310+03	.269+03
.320+03	.269+03
.330+03	.269+03
.340+03	.269+03
.350+03	.269+03
.360+03	.269+03
.370+03	.269+03
.380+03	.269+03

```
   IOP        IOR        IOS        IOO        IOO
    0          1          1          0          1
  ALTRU       PIP2
 .100+00     .100+01
NFM=  5    NFMRD=  0


FO/F- =   .500+01    EN=   .129+01
FO/F- =   .100+02    EN=   .132+01
FO/F- =   .200+02    EN=   .135+01
FO/F- =   .400+02    EN=   .138+01
FO/F- =   .800+02    EN=   .142+01
```

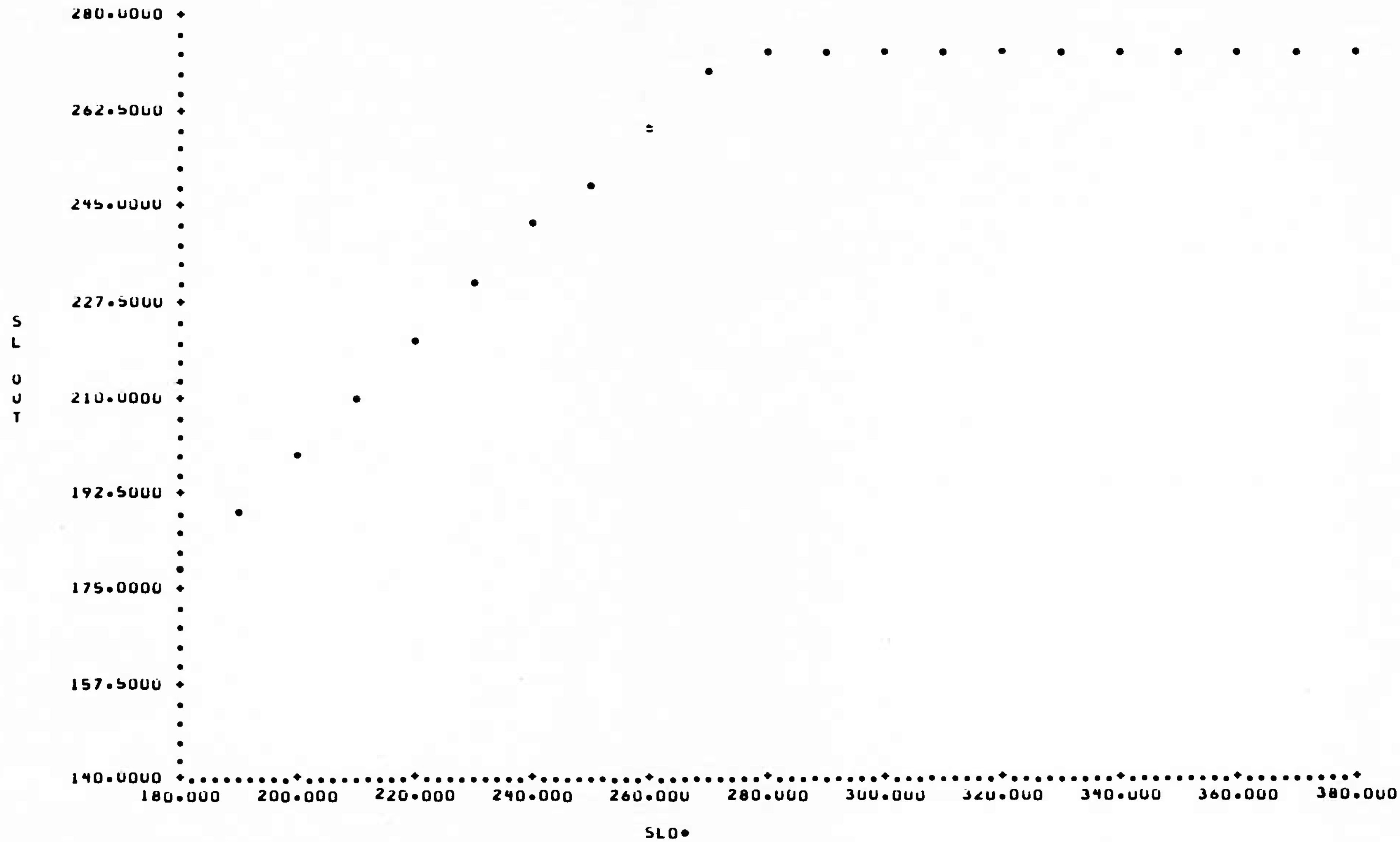


ALPHA(T)*RO= .100+00 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS



ALPHA(T)*RO=   .100+00 DB
CURVE 1 IS FOR FO/F-=   .500+01
CURVE 2 IS FOR FO/F-=   .100+02
CURVE 3 IS FOR FO/F-=   .200+02
CURVE 4 IS FOR FO/F-=   .400+02
CURVE 5 IS FOR FO/F-=   .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

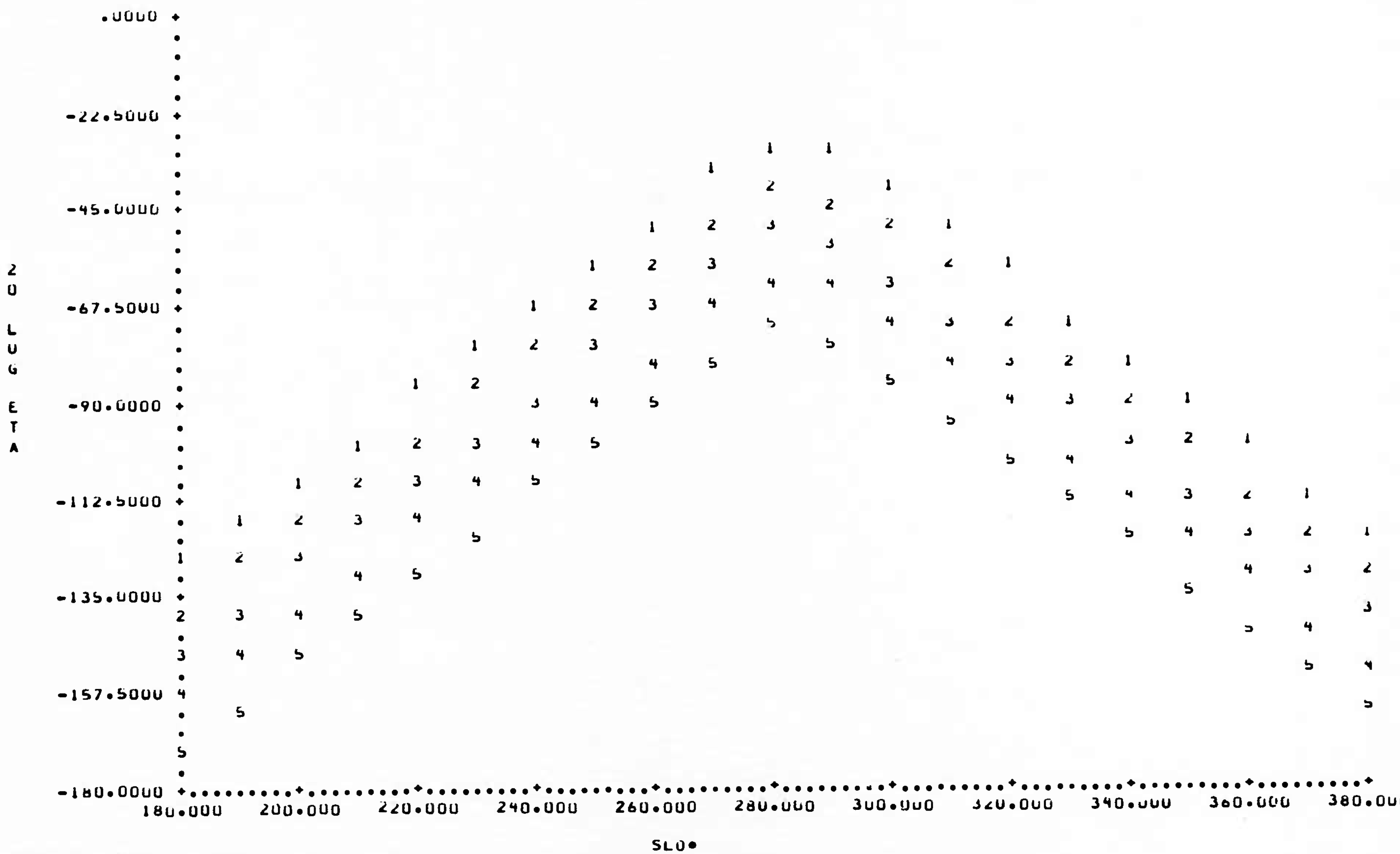
SL0.	SL-(1).	SL-(2).	SL-(3).	SL-(4).	SL-(5).
.180+03	.618+02	.535+02	.448+02	.356+02	.259+02
.190+03	.818+02	.735+02	.648+02	.556+02	.459+02
.200+03	.102+03	.935+02	.848+02	.756+02	.659+02
.210+03	.122+03	.114+03	.105+03	.956+02	.859+02
.220+03	.142+03	.134+03	.125+03	.116+03	.106+03
.230+03	.162+03	.154+03	.145+03	.136+03	.126+03
.240+03	.182+03	.174+03	.165+03	.156+03	.146+03
.250+03	.202+03	.194+03	.185+03	.176+03	.166+03
.260+03	.222+03	.213+03	.205+03	.195+03	.186+03
.270+03	.242+03	.232+03	.223+03	.214+03	.204+03
.280+03	.252+03	.244+03	.235+03	.226+03	.216+03
.290+03	.255+03	.247+03	.238+03	.229+03	.219+03
.300+03	.256+03	.247+03	.239+03	.230+03	.220+03
.310+03	.256+03	.248+03	.239+03	.230+03	.220+03
.320+03	.256+03	.248+03	.239+03	.230+03	.220+03
.330+03	.256+03	.248+03	.239+03	.230+03	.220+03
.340+03	.256+03	.248+03	.239+03	.230+03	.220+03
.350+03	.256+03	.248+03	.239+03	.230+03	.220+03
.360+03	.256+03	.248+03	.239+03	.230+03	.220+03
.370+03	.256+03	.248+03	.239+03	.230+03	.220+03
.380+03	.256+03	.248+03	.239+03	.230+03	.220+03



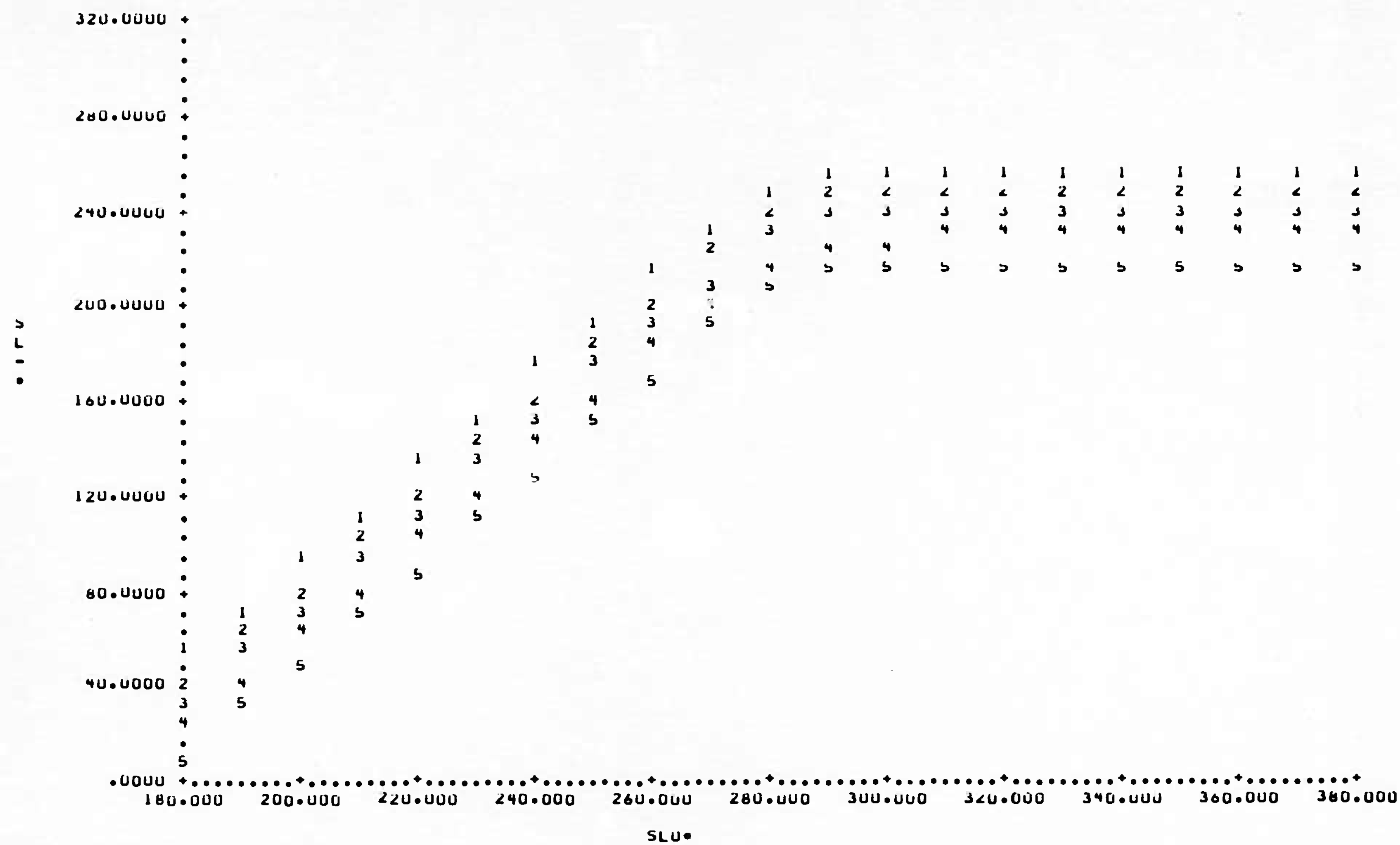
ALPHA(O)*RO=   .577-02 NEPERS
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

SLO*	SL OUT
.180+03	.180+03
.190+03	.190+03
.200+03	.200+03
.210+03	.210+03
.220+03	.220+03
.230+03	.230+03
.240+03	.240+03
.250+03	.250+03
.260+03	.260+03
.270+03	.268+03
.280+03	.272+03
.290+03	.272+03
.300+03	.273+03
.310+03	.273+03
.320+03	.273+03
.330+03	.273+03
.340+03	.273+03
.350+03	.273+03
.360+03	.273+03
.370+03	.273+03
.380+03	.273+03

```
   IOP          IOE          IOS          IOD          IOO
    0            1            1            0            1
  ALTRO         PIP2
  .100+01      .100+01
NFM=  5    NFMRD=  0


FO/F- =    .500+01    EN=    .150+01
FO/F- =    .100+02    EN=    .155+01
FO/F- =    .200+02    EN=    .159+01
FO/F- =    .400+02    EN=    .164+01
FO/F- =    .800+02    EN=    .168+01
```

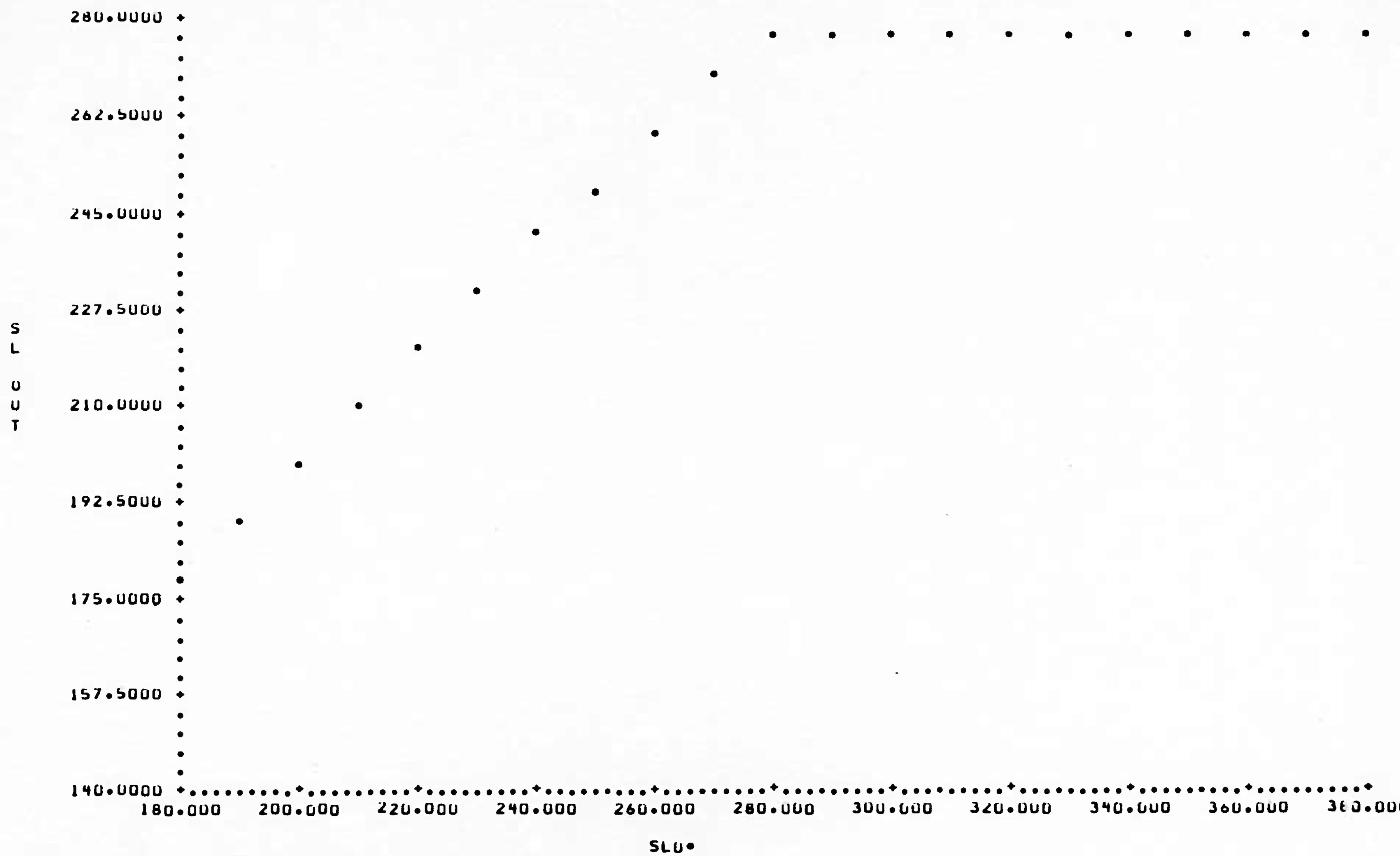


ALPHA(T)*RO=   .100+01 DB
CURVE 1 IS FOR FO/F-=   .500+01
CURVE 2 IS FOR FO/F-=   .100+02
CURVE 3 IS FOR FO/F-=   .200+02
CURVE 4 IS FOR FO/F-=   .400+02
CURVE 5 IS FOR FO/F-=   .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS



ALPHA(T)•RU=   .100+01 DB
CURVE 1 IS FOR FO/F-=   .500+01
CURVE 2 IS FOR FO/F-=   .100+02
CURVE 3 IS FOR FO/F-=   .200+02
CURVE 4 IS FOR FO/F-=   .400+02
CURVE 5 IS FOR FO/F-=   .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

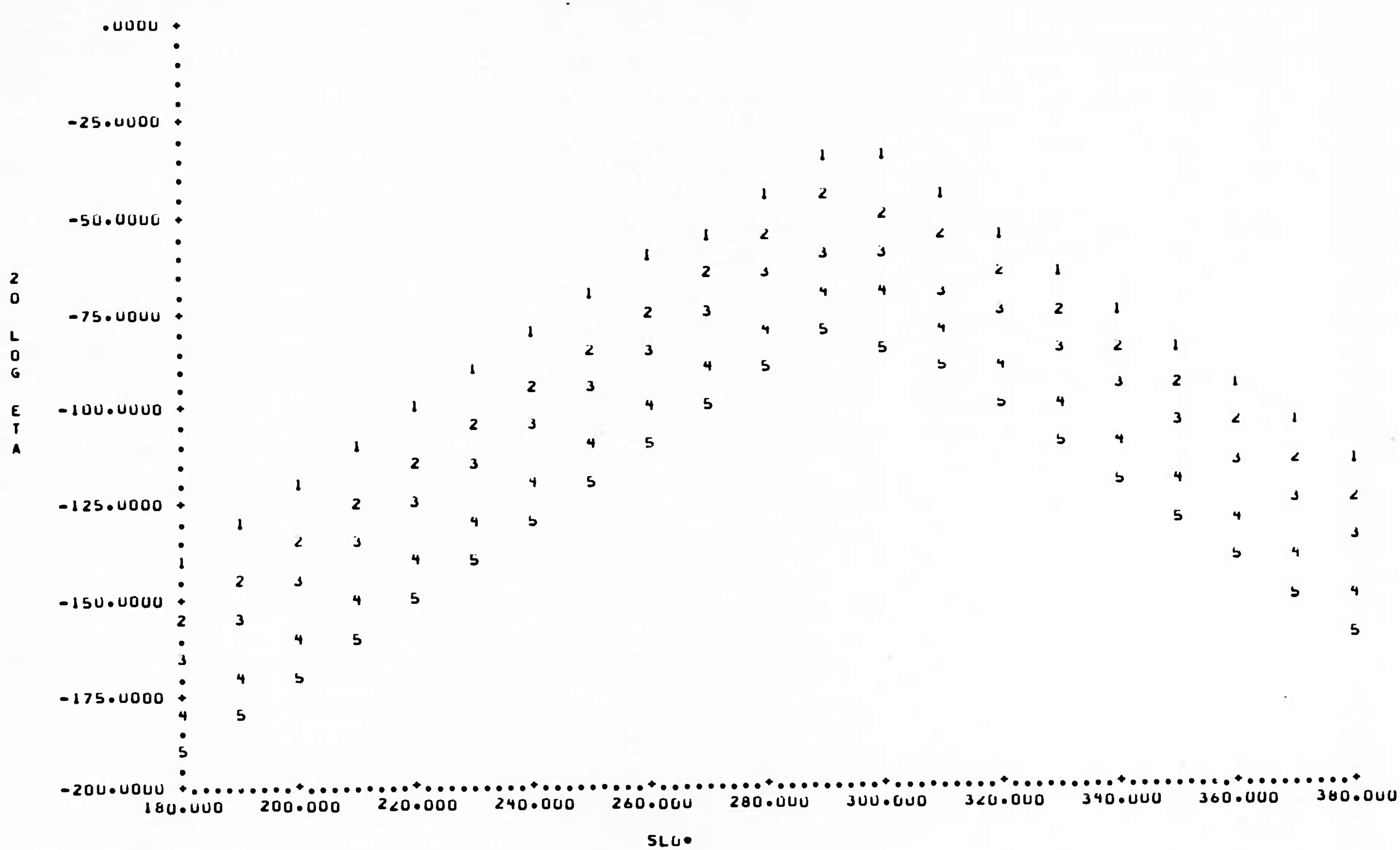
SLO.	SL-(1).	SL-(2).	SL-(3).	SL-(4).	SL-(5).
.180+03	.525+02	.426+02	.321+02	.212+02	.977+01
.190+03	.725+02	.626+02	.521+02	.412+02	.298+02
.200+03	.925+02	.826+02	.721+02	.612+02	.498+02
.210+03	.113+03	.103+03	.921+02	.812+02	.698+02
.220+03	.133+03	.123+03	.112+03	.101+03	.898+02
.230+03	.153+03	.143+03	.132+03	.121+03	.110+03
.240+03	.173+03	.163+03	.152+03	.141+03	.130+03
.250+03	.193+03	.183+03	.172+03	.161+03	.150+03
.260+03	.213+03	.203+03	.192+03	.181+03	.170+03
.270+03	.232+03	.222+03	.212+03	.201+03	.189+03
.280+03	.249+03	.239+03	.229+03	.218+03	.206+03
.290+03	.257+03	.247+03	.237+03	.226+03	.214+03
.300+03	.259+03	.249+03	.238+03	.227+03	.216+03
.310+03	.259+03	.250+03	.239+03	.228+03	.217+03
.320+03	.259+03	.250+03	.239+03	.228+03	.217+03
.330+03	.259+03	.250+03	.239+03	.228+03	.217+03
.340+03	.259+03	.250+03	.239+03	.228+03	.217+03
.350+03	.259+03	.250+03	.239+03	.228+03	.217+03
.360+03	.259+03	.250+03	.239+03	.228+03	.217+03
.370+03	.259+03	.250+03	.239+03	.228+03	.217+03
.380+03	.259+03	.250+03	.239+03	.228+03	.217+03



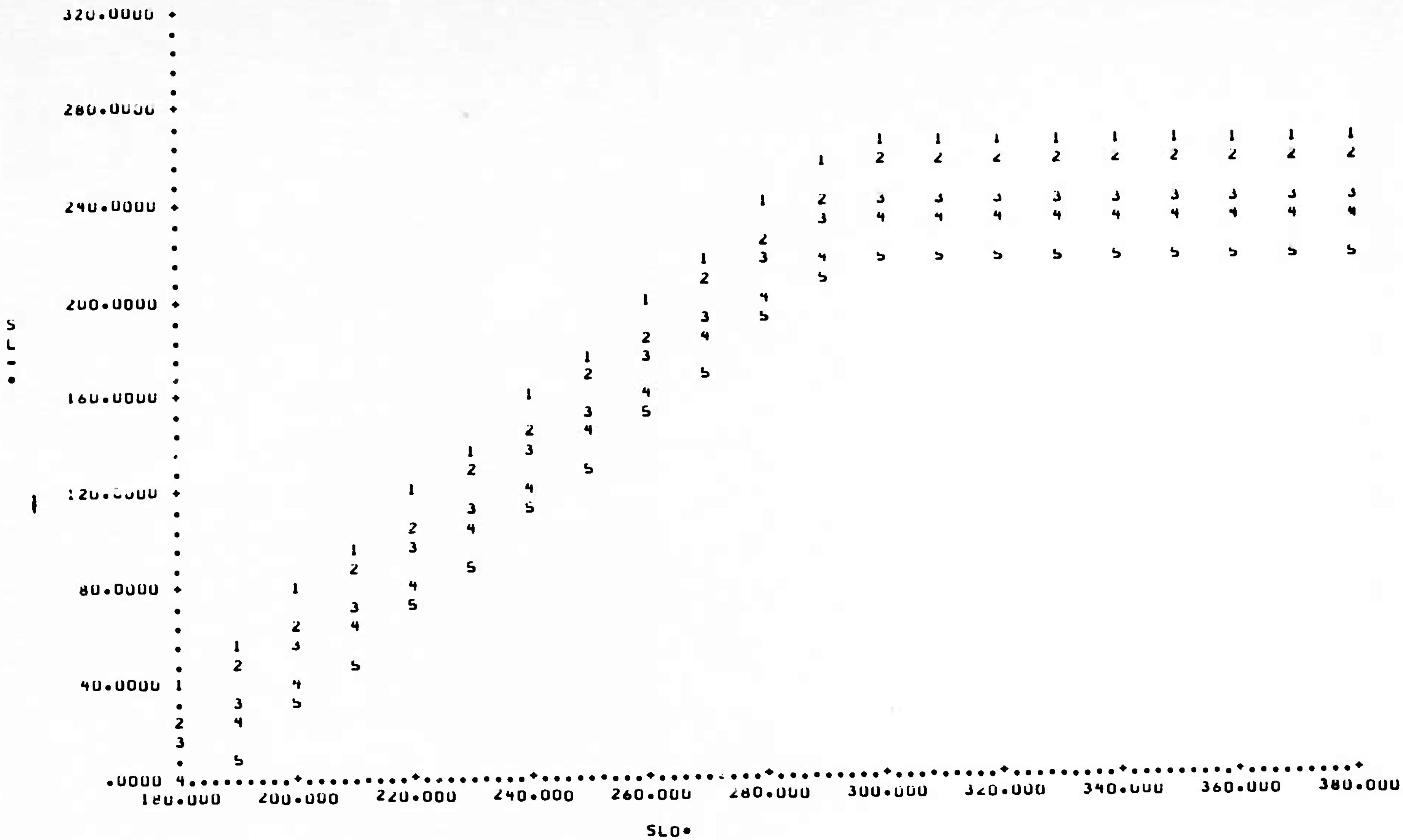
ALPHA(0)*RO= .577-01 NEPERS
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

SLO.	SL OUT
.180+03	.180+03
.190+03	.190+03
.200+03	.200+03
.210+03	.210+03
.220+03	.220+03
.230+03	.230+03
.240+03	.240+03
.250+03	.250+03
.260+03	.260+03
.270+03	.269+03
.280+03	.276+03
.290+03	.278+03
.300+03	.278+03
.310+03	.278+03
.320+03	.278+03
.330+03	.278+03
.340+03	.278+03
.350+03	.278+03
.360+03	.278+03
.370+03	.278+03
.380+03	.278+03

```
   IOP        IOE        IOS        IOD        IOO
    0          1          1          0          1
  ALTRO       PIP2
  .100+02     .100+01
NFM=  5    NFMRD=  0


FO/F- =   .500+01    EN=   .179+01
FO/F- =   .100+02    EN=   .183+01
FO/F- =   .200+02    EN=   .186+01
FO/F- =   .400+02    EN=   .188+01
FO/F- =   .800+02    EN=   .189+01
```



ALPHA(T)*RO= .100+02 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

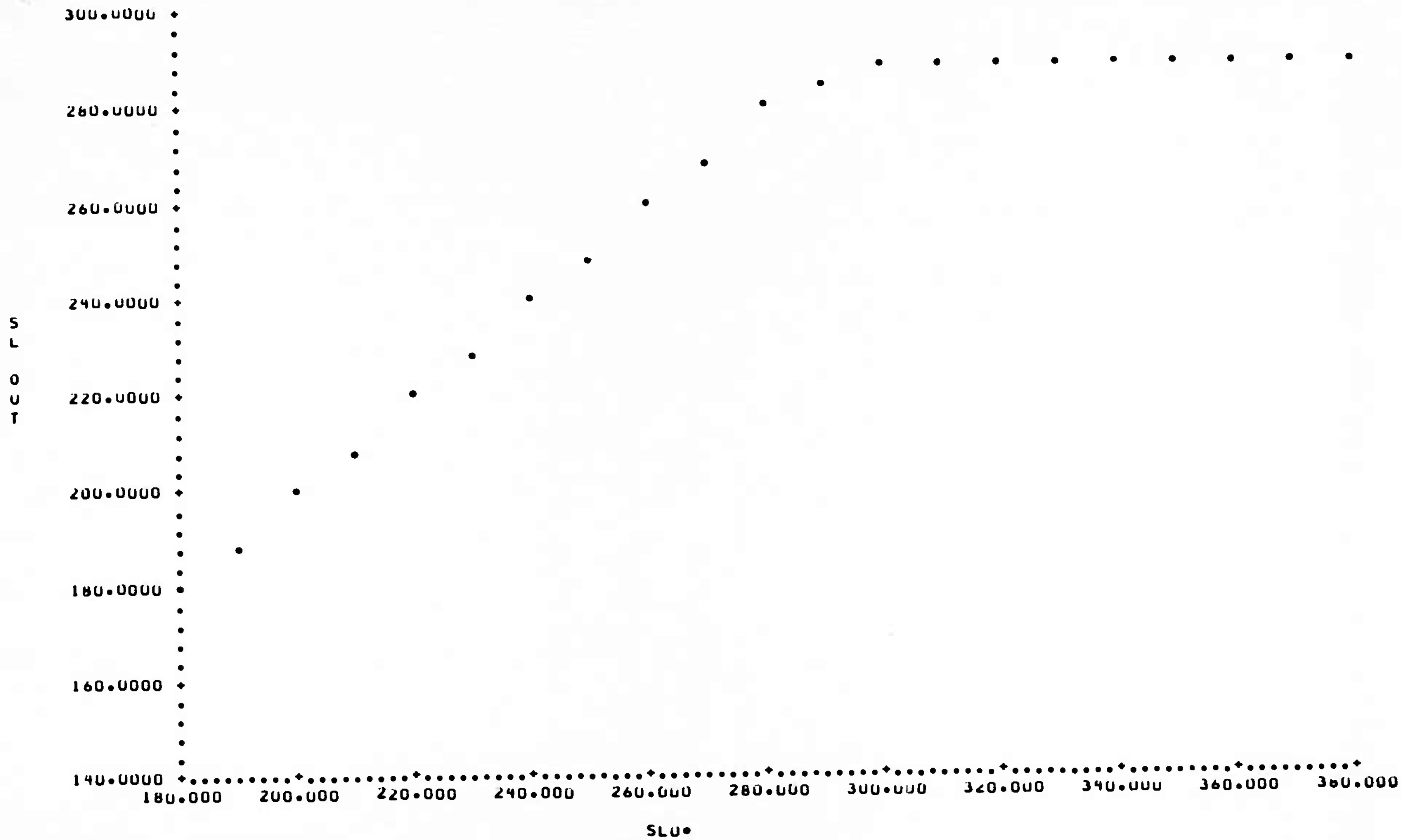


••••• THE FOLLOWING 1 POINTS WERE OUT OF RANGE •••••

X= 180.000 Y= -9.497

ALPHA(T)•RO= .100+02 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

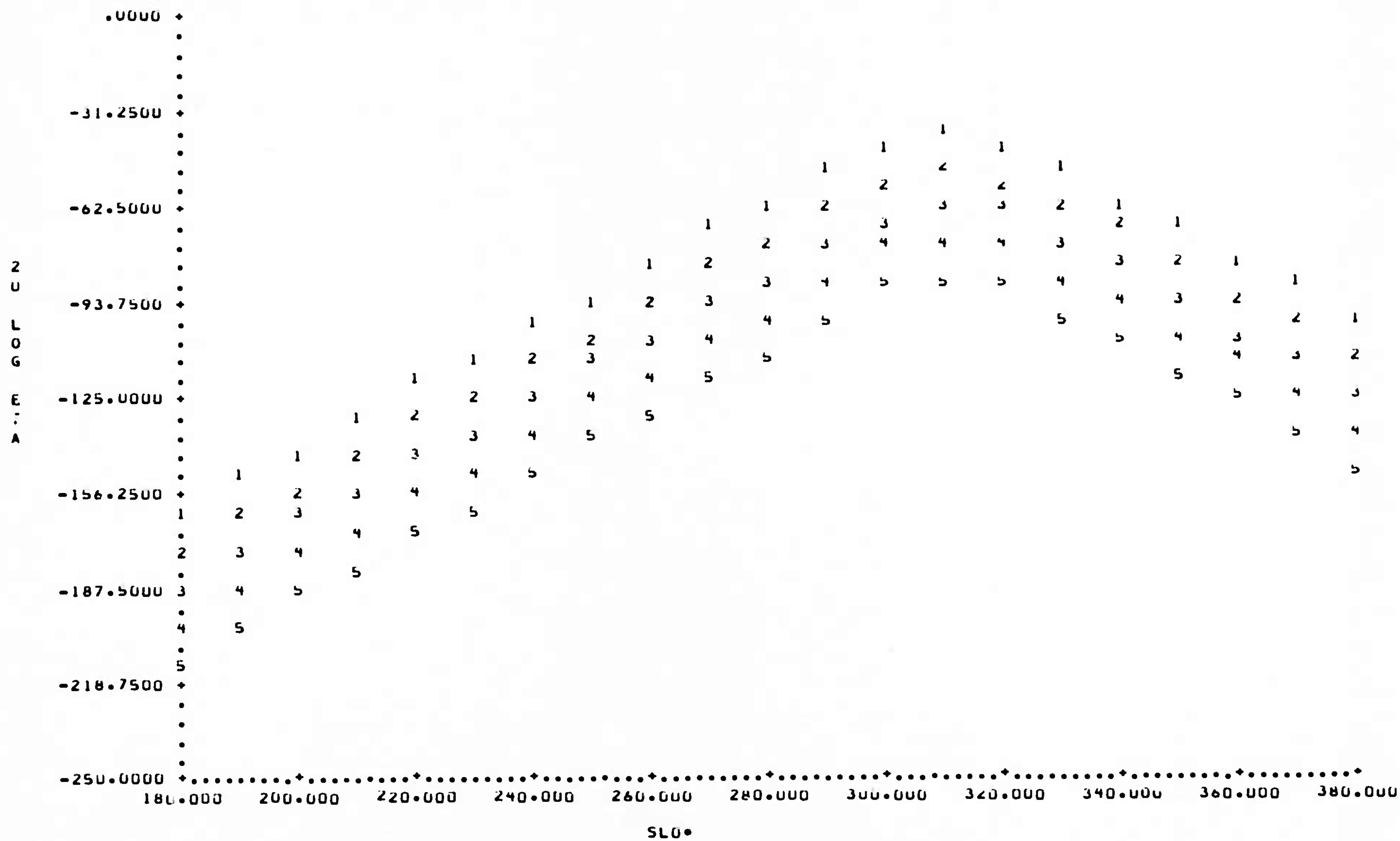
SLO.	SL-(1).	SL-(2).	SL-(3).	SL-(4).	SL-(5).
.180+03	.375+02	.260+02	.143+02	.245+01	-.950+01
.190+03	.575+02	.460+02	.343+02	.225+02	.105+02
.200+03	.775+02	.660+02	.543+02	.425+02	.305+02
.210+03	.975+02	.860+02	.743+02	.625+02	.505+02
.220+03	.118+03	.106+03	.943+02	.825+02	.705+02
.230+03	.138+03	.126+03	.114+03	.102+03	.905+02
.240+03	.158+03	.146+03	.134+03	.122+03	.111+03
.250+03	.178+03	.166+03	.154+03	.142+03	.131+03
.260+03	.198+03	.186+03	.174+03	.162+03	.150+03
.270+03	.217+03	.206+03	.194+03	.182+03	.170+03
.280+03	.237+03	.226+03	.214+03	.202+03	.190+03
.290+03	.255+03	.243+03	.232+03	.220+03	.208+03
.300+03	.264+03	.252+03	.240+03	.229+03	.217+03
.310+03	.266+03	.254+03	.242+03	.230+03	.218+03
.320+03	.266+03	.255+03	.243+03	.231+03	.219+03
.330+03	.266+03	.255+03	.243+03	.231+03	.219+03
.340+03	.266+03	.255+03	.243+03	.231+03	.219+03
.350+03	.266+03	.255+03	.243+03	.231+03	.219+03
.360+03	.266+03	.255+03	.243+03	.231+03	.219+03
.370+03	.266+03	.255+03	.243+03	.231+03	.219+03
.380+03	-266+03	.255+03	.243+03	.231+03	.219+03



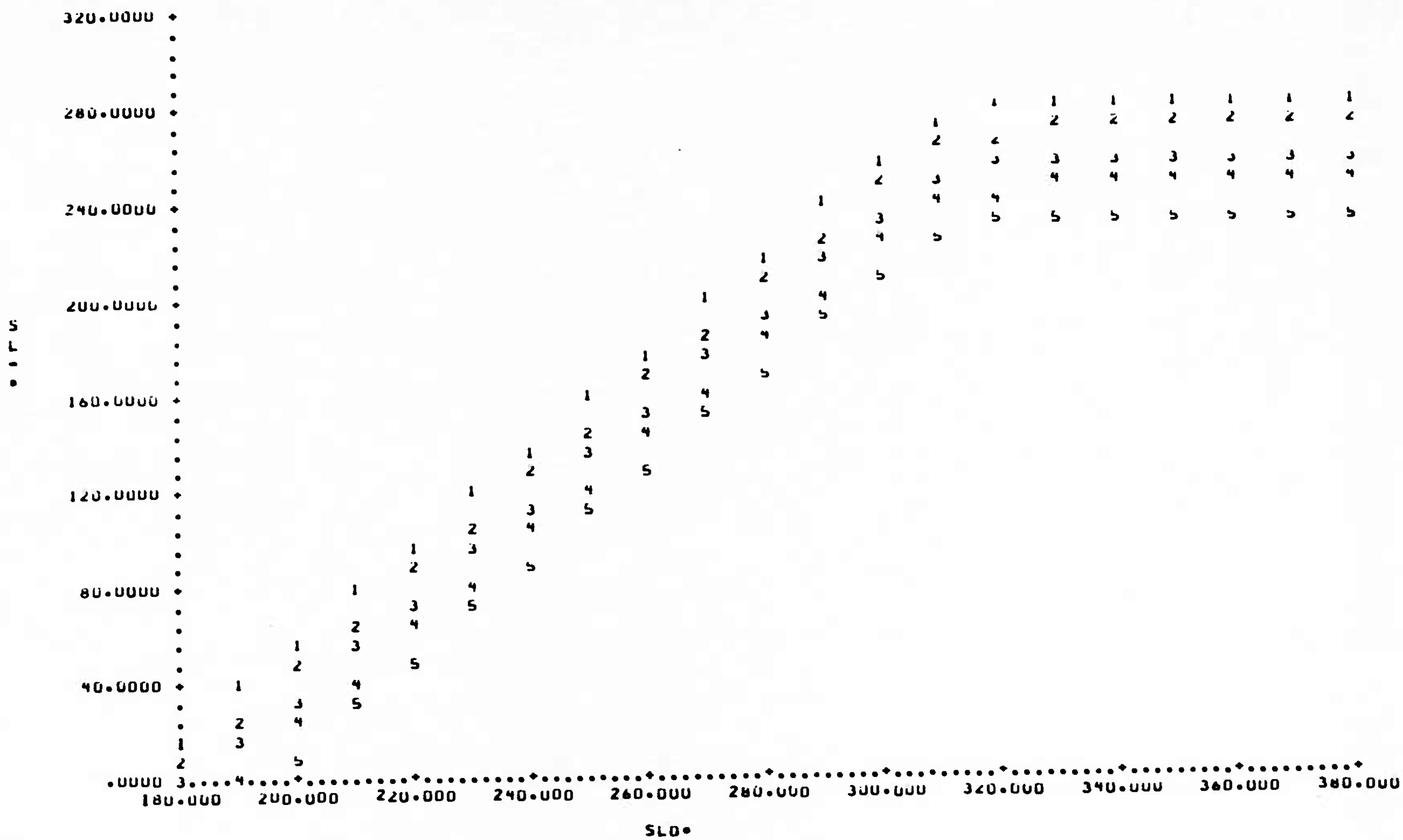
ALPHA(O)•RO= .577+00 NEPERS
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

SLO.	SL OUT
.180+03	.180+03
.190+03	.190+03
.200+03	.200+03
.210+03	.210+03
.220+03	.220+03
.230+03	.230+03
.240+03	.240+03
.250+03	.250+03
.260+03	.260+03
.270+03	.270+03
.280+03	.279+03
.290+03	.285+03
.300+03	.287+03
.310+03	.287+03
.320+03	.287+03
.330+03	.287+03
.340+03	.287+03
.350+03	.287+03
.360+03	.287+03
.370+03	.287+03
.380+03	.287+03

```
    IOP          IOE          IOS          IOD          IOO
     0            1            1            0            1
   ALTRU         PIP2
  .100+03       .100+01
 NFM=  5    NFMRD=  0



 FO/F- =    .500+01    EN=    .196+01
 FO/F- =    .100+02    EN=    .197+01
 FO/F- =    .200+02    EN=    .198+01
 FO/F- =    .400+02    EN=    .198+01
 FO/F- =    .600+02    EN=    .198+01
```



ALPHA(T)*RO= .100+03 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

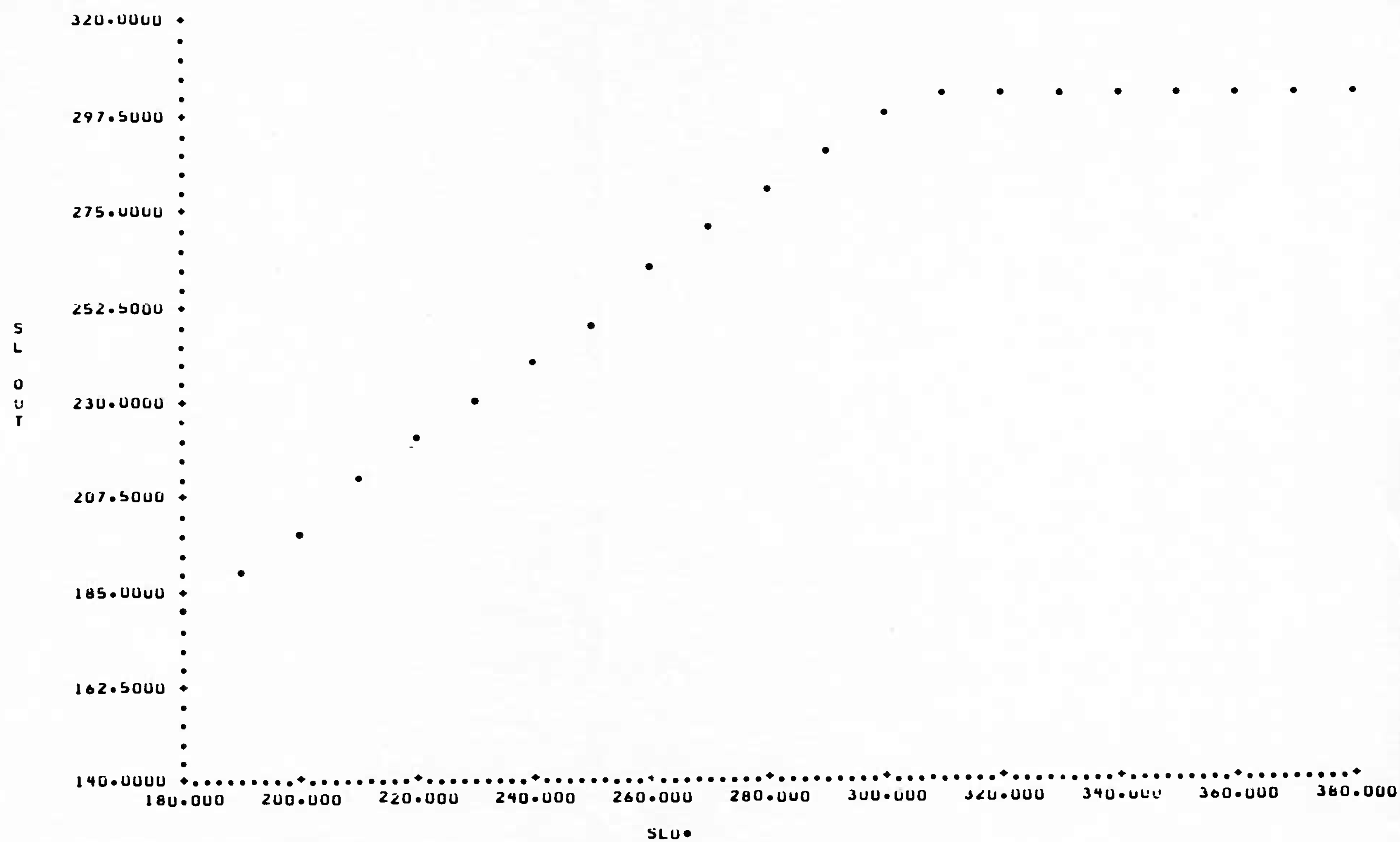


..... THE FOLLOWING 3 POINTS WERE OUT OF RANGE .....

X= 180.000 Y= -17.362
X= 180.000 Y= -29.414
X= 190.000 Y= -9.414

ALPHA(T)*RO= .100+03 DB
CURVE 1 IS FOR FO/F-= .500+01
CURVE 2 IS FOR FO/F-= .100+02
CURVE 3 IS FOR FO/F-= .200+02
CURVE 4 IS FOR FO/F-= .400+02
CURVE 5 IS FOR FO/F-= .800+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

SLU•	SL-(1)•	SL-(2)•	SL-(3)•	SL-(4)•	SL-(5)•
.180+03	.186+02	.664+01	-.536+01	-.174+02	-.294+02
.190+03	.386+02	.266+02	.146+02	.262+01	-.941+01
.200+03	.586+02	.466+02	.346+02	.226+02	.106+02
.210+03	.786+02	.666+02	.546+02	.426+02	.306+02
.220+03	.986+02	.866+02	.746+02	.626+02	.506+02
.230+03	.119+03	.107+03	.946+02	.826+02	.706+02
.240+03	.139+03	.127+03	.115+03	.103+03	.906+02
.250+03	.159+03	.147+03	.135+03	.123+03	.111+03
.260+03	.179+03	.167+03	.155+03	.143+03	.131+03
.270+03	.199+03	.187+03	.175+03	.163+03	.151+03
.280+03	.219+03	.207+03	.195+03	.183+03	.171+03
.290+03	.239+03	.227+03	.215+03	.203+03	.191+03
.300+03	.258+03	.246+03	.234+03	.222+03	.210+03
.310+03	.273+03	.261+03	.249+03	.237+03	.225+03
.320+03	.279+03	.267+03	.255+03	.243+03	.231+03
.330+03	.280+03	.268+03	.256+03	.244+03	.232+03
.340+03	.281+03	.269+03	.257+03	.245+03	.232+03
.350+03	.281+03	.269+03	.257+03	.245+03	.232+03
.360+03	.281+03	.269+03	.257+03	.245+03	.232+03
.370+03	.281+03	.269+03	.257+03	.245+03	.232+03
.380+03	.281+03	.269+03	.257+03	.245+03	.232+03



ALPHA(0)*RO= .577+01 NEPERS
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

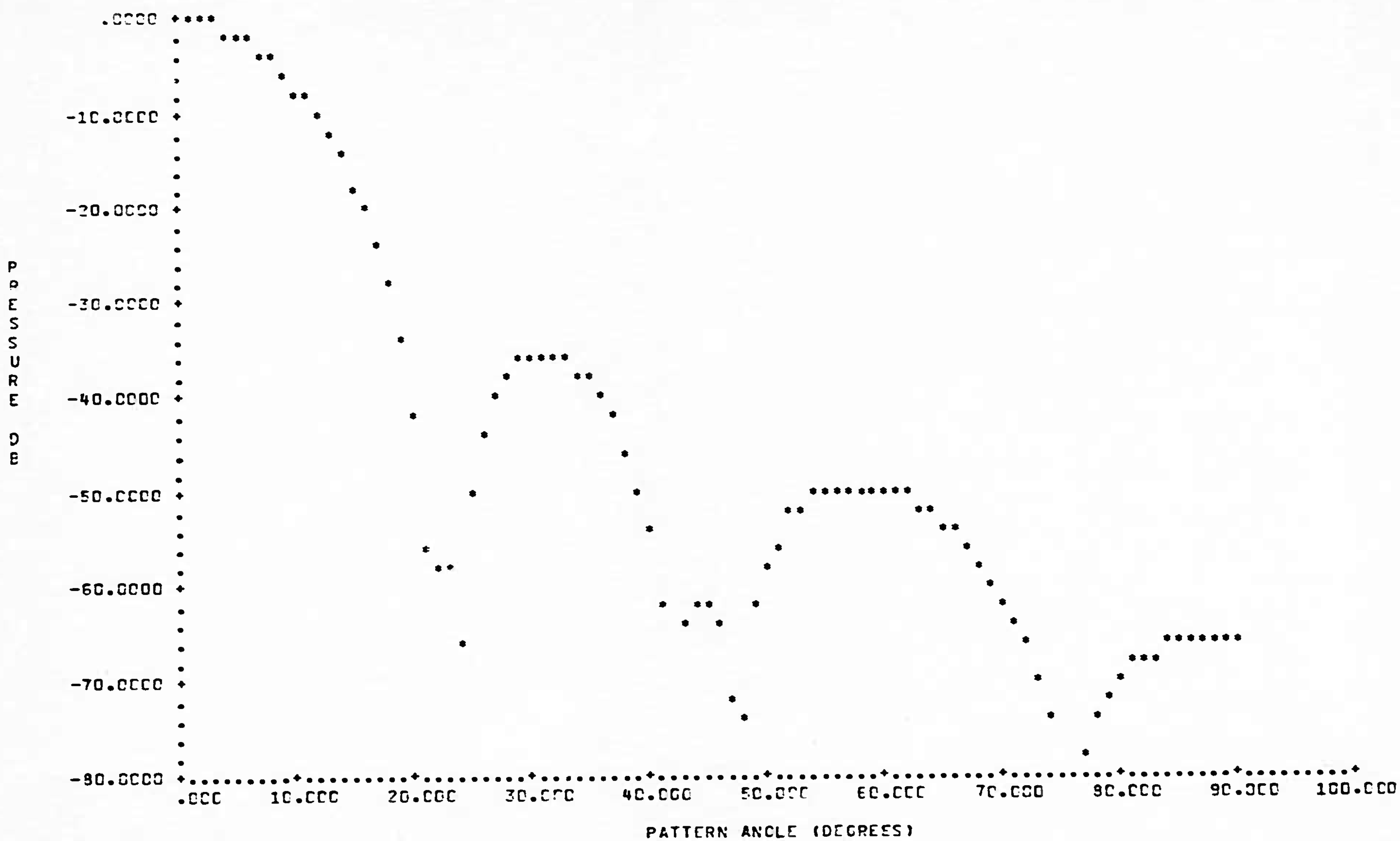
SLU•	SL OUT
•180+03	•180+03
•190+03	•190+03
•200+03	•200+03
•210+03	•210+03
•220+03	•220+03
•230+03	•230+03
•240+03	•240+03
•250+03	•250+03
•260+03	•260+03
•270+03	•270+03
•280+03	•280+03
•290+03	•290+03
•300+03	•298+03
•310+03	•301+03
•320+03	•302+03
•330+03	•302+03
•340+03	•302+03
•350+03	•302+03
•360+03	•302+03
•370+03	•302+03
•380+03	•302+03

@FIN

FIGURE 2

(Pages 87 to 99)

```
@XQT SONIC*NONLN.BEAM
 NOC=  7

    IOP          IOE          IOS          IOC          IOO
     1            C            C            C            C
   ALTRC        P1P2
  .10C-04       .10C+01
 NFM=  1    NFMRD=  1
   FC/FM=       .10C+02
 LIN=  0    IOC=  1
 KC*A=   .10C+02
 PATTERN SLS=   .24C+03    FC/F-=    .10C+02    C3=  -.30C+02    NP= 9C    TD=   .10C+01


 FC/F- =   .10C+02    EN=   .1C3+01
```

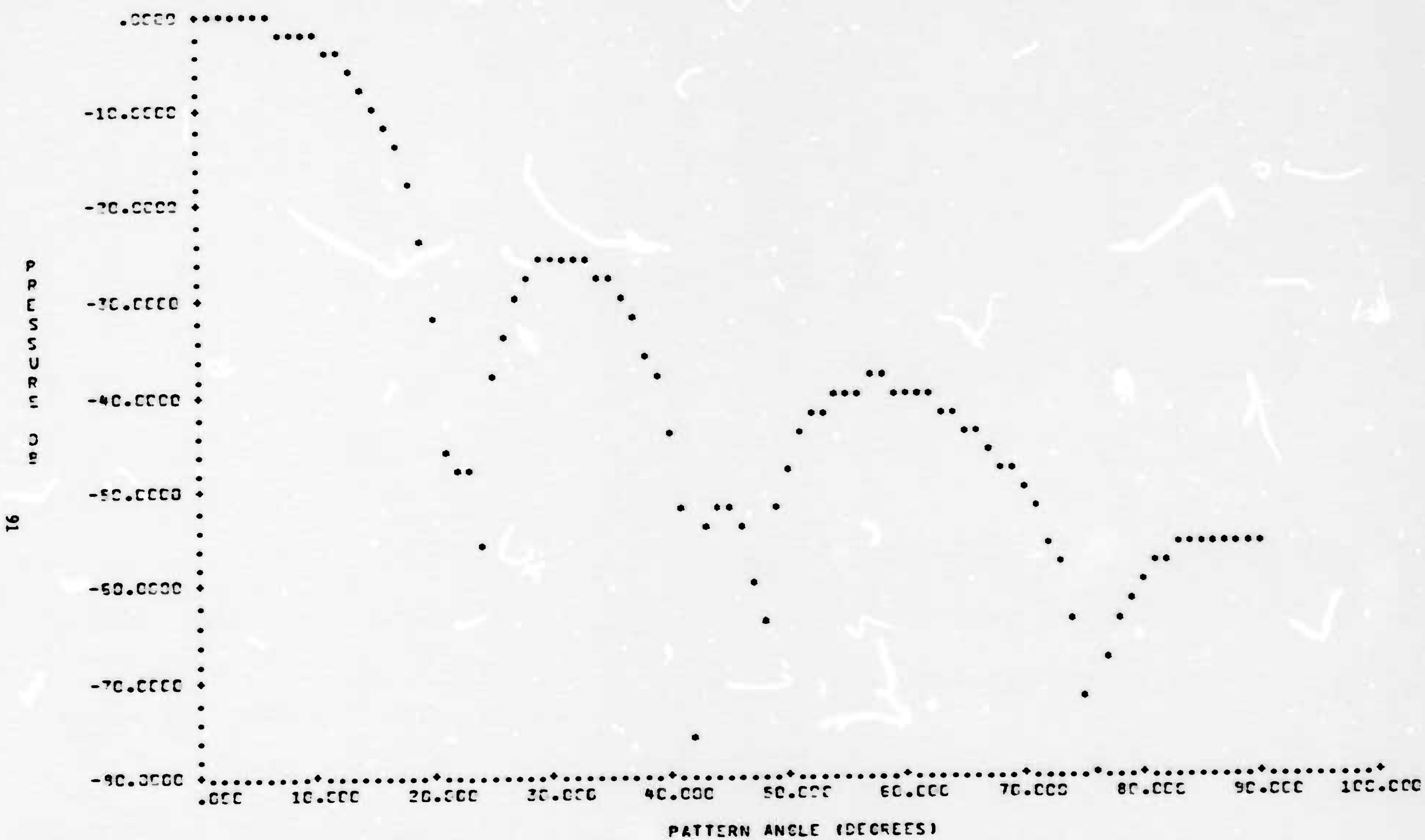


***** THE FOLLOWING 3 POINTS WERE OUT OF RANGE *****

X=	42.000	Y=	-85.437
X=	75.000	Y=	-92.565
X=	76.000	Y=	-89.510

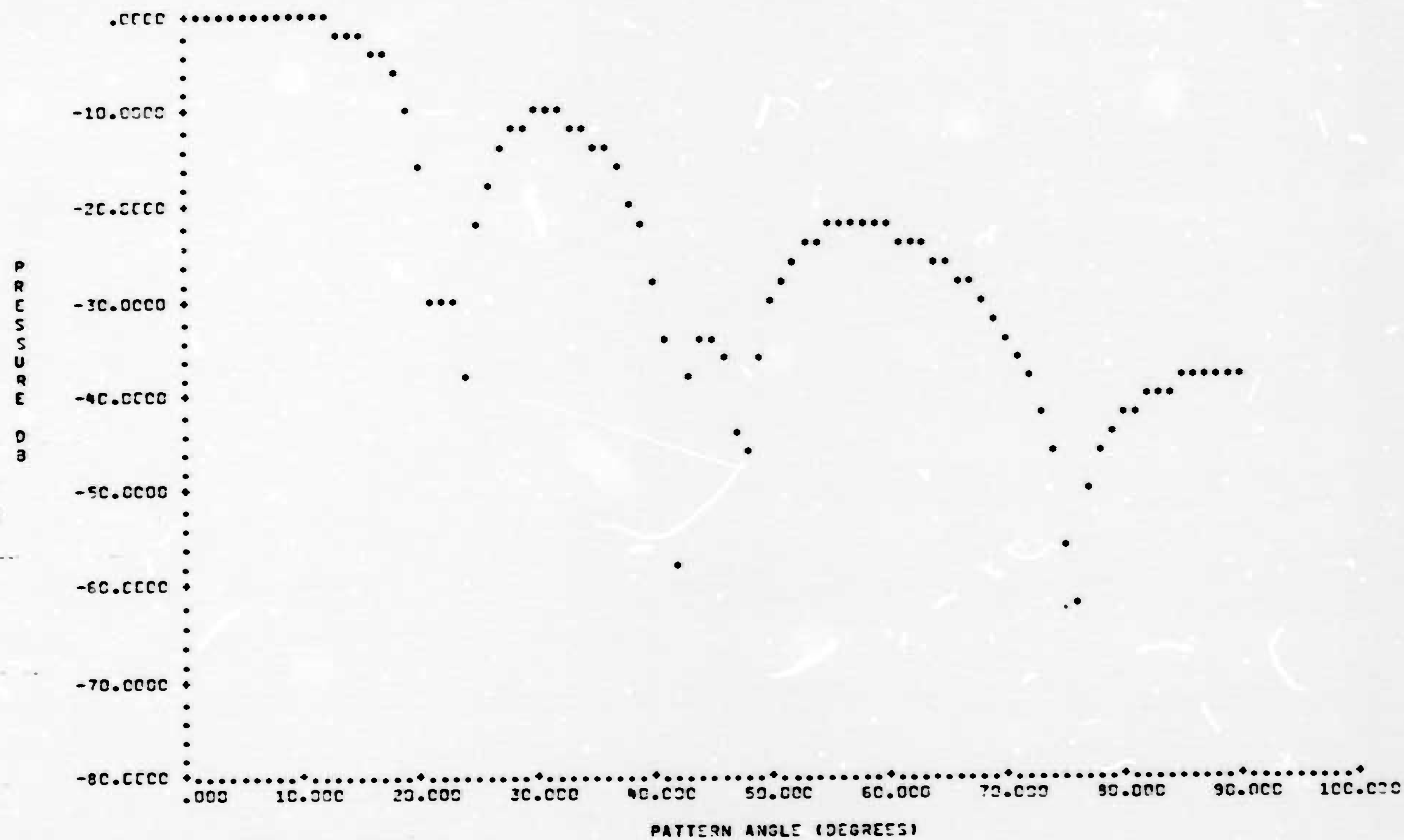
SLO*= .240+03 FO/F-= .100+02
ALPHA(T)*RO= .100-04 DB
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

```
    IOP           IOC           IOC           IOC           IOC
     1             0             0             0             0
   ALTRO          P1P2
  .100-04        .100+01
NFM=  1    NFMRD=  1
   FC/FM=        .100+02
LIN=  0    IOC=  1
KD*A=   .100+02
PATTERN SLS=   .270+03    FC/F-=   .100+02    DB=  -.300+02    NP= 90    TD=   .100+01


FC/F- =   .100+02    EN=   .103+01
```



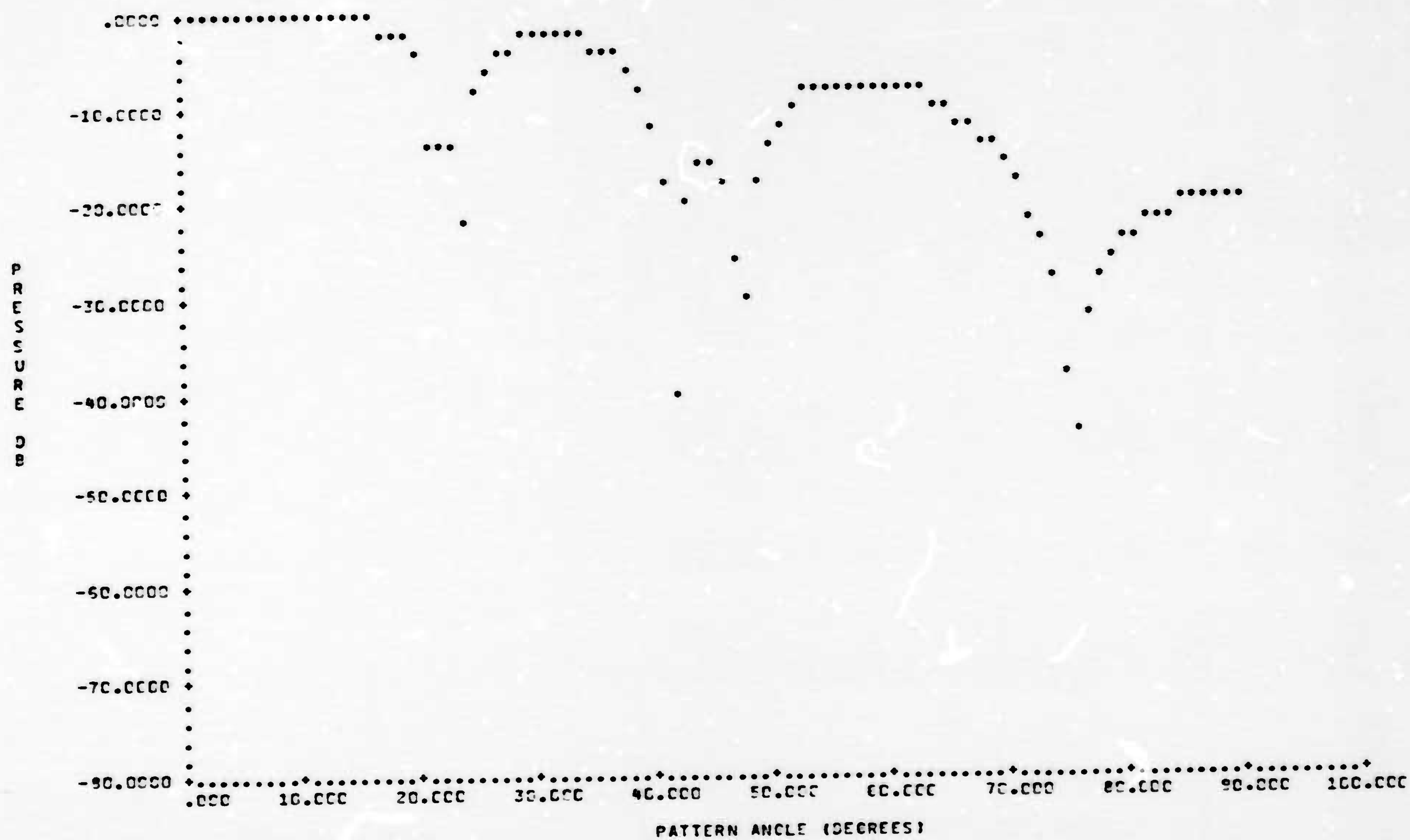
SLC•= .270+03 FC/F-= .100+02
ALPHA(T)•RE= .100-04 DB
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

```
    ICP        ICC        ICS        IOC        ICO
     1          0          0          0          0
   ALTRC       PIP2
  .100-04     .100+01
NFM=  1    NFMRD=  1
  FC/FM=      .100+02
LIN=  0    IOC=  1
KC*A=   .100+02
PATTERN SLS=   .280+03    FC/F-=   .100+02    DB=  -.800+02    NP= 90    TC=   .100+01


FC/F- =   .100+02    EN=   .108+01
```



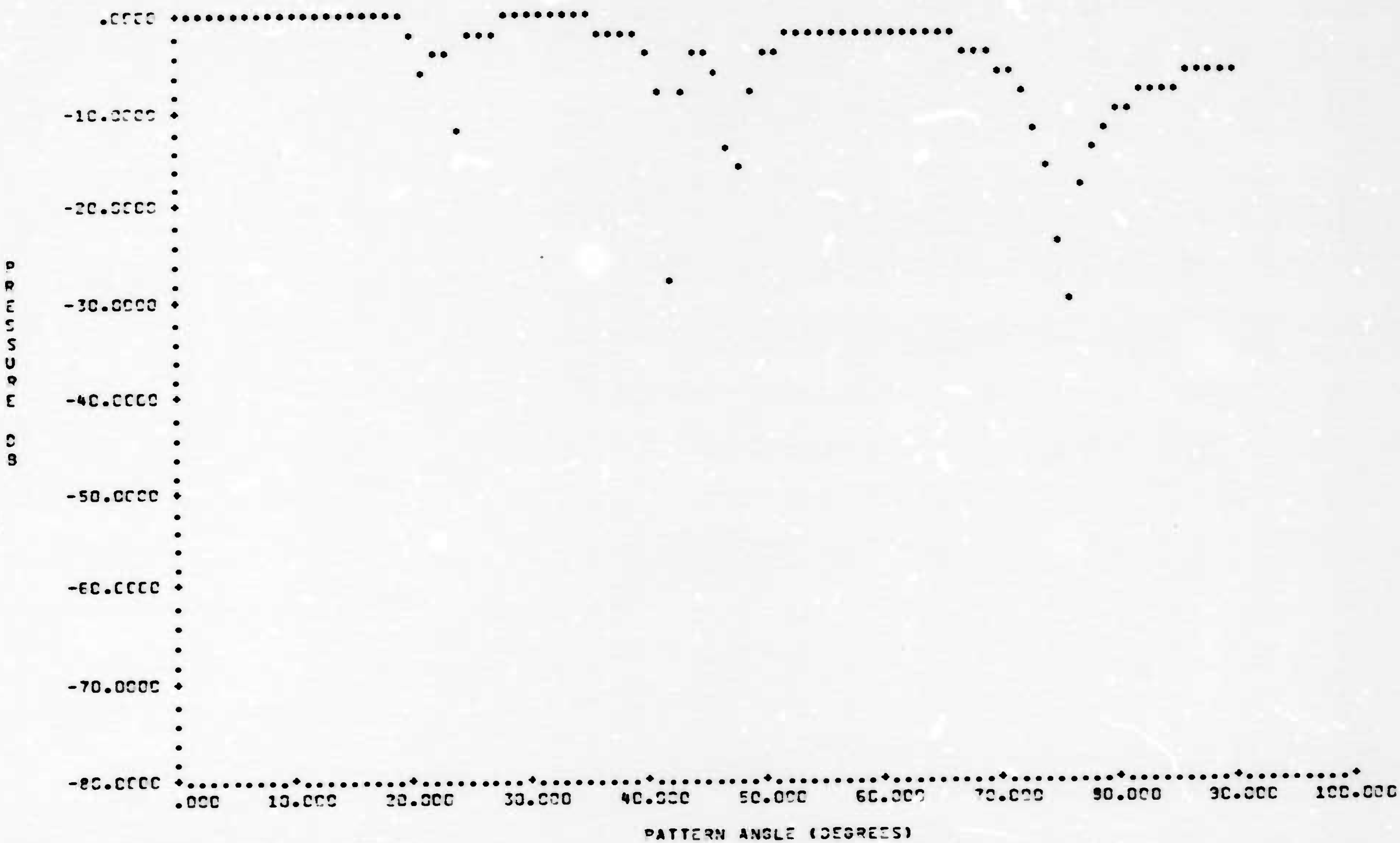
SLC•= .280+C3 FC/F-= .100+C2
ALPHA(T)•RC= .100-04 DB
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

```
    ICP        ICE        IOS        IOD        IOC
     1          C          0          0          0
   ALTRO       P1P2
  .100-04     .100+01
NFM=  1   NFMRO=  1
   FC/FM=     .100+02
LIN=  0   IOC=  1
KC*A=   .100+02
PATTERN SLS=   .290+03    FC/F-=   .100+02    D3=  -.300+02    NP= 90    TD=   .100+01


FC/F- =   .100+02    EN=   .103+01
```



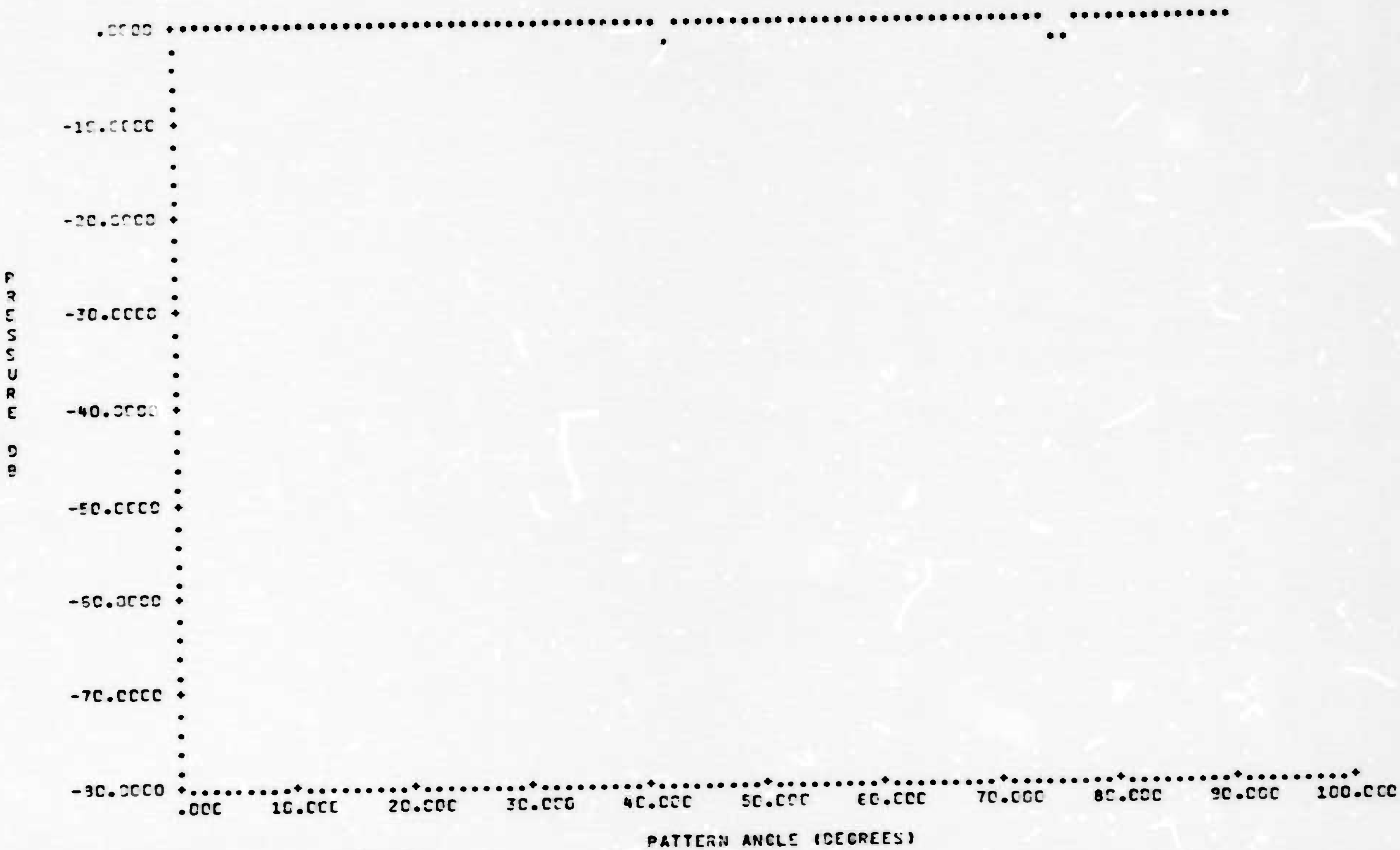
SLO•= .290+03 FC/F-= .100+02
ALPHA(T)•RC= .100-04 DB
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

```
    IOP          ICE          ICC          IOC          ICC
     1            C            C            0            C
   ALTRC        P1P2
  .100-04      .100+01
NFM=  1    NFMPD=  1
   FD/FM=      .100+02
LIN=  0    IOC=  1
KC*A=   .100+02
PATTERN SLS=   .300+03    FC/F-=   .100+02    DB=  -.800+02    NP= 90    TD=   .100+01


FC/F- =   .100+02    EN=   .108+01
```



SLC•= .300+03 FC/F-= .100+02
ALPHA(T)•RC= .10L-04 03
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

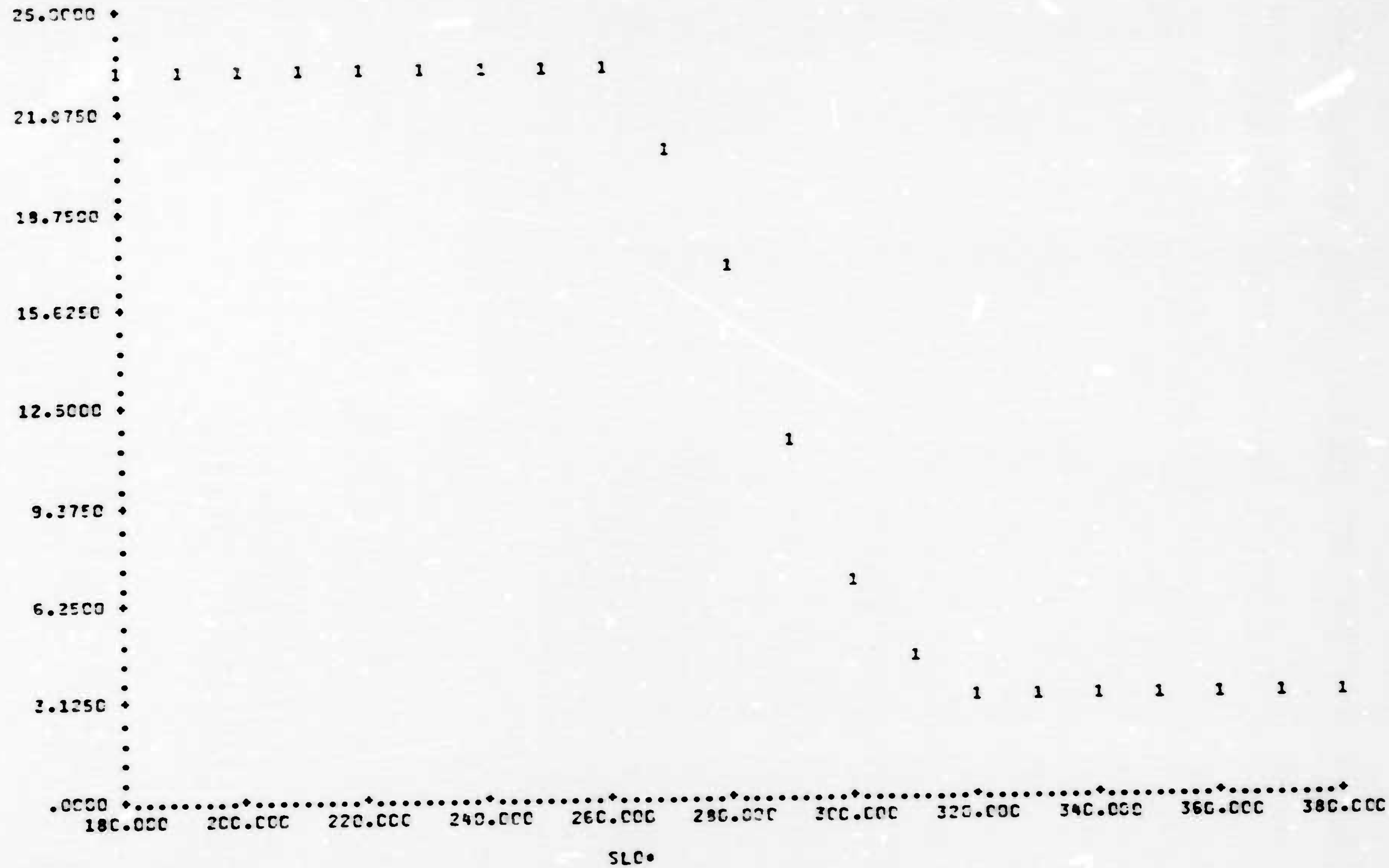
```
    IOP        IOE        IOS        IOD        IOO
     1          C          0          C          0
   ALTRO       P1P2
   .100-04     .100+01
NFM=  1    NFMRD=  1
   FO/FM=      .100+02
LIN=  0    IOC=  1
KC*A=   .100+02
PATTERN SLS=   .330+03    FO/F-=   .100+02    DB=  -.800+02    NP= 30    TD=   .100+01


FO/F- =   .100+02    EN=   .103+01
```



SLC*= .330+03 FC/F-= .100+02
ALPHA(T)*RC= .100-04 DB
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

FIGURE 3

(Pages 101 to 104)



ALPHA(T)•RC= .100-04 03
CURVE 1 IS FOR FC/F-= .100+02
PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS

SLC.	DI(1)	DI(2)	DI(3)	DI(4)	DI(5)
.180+03	.234+02				
.190+03	.234+02				
.200+03	.234+02				
.210+03	.234+02				
.220+03	.234+02				
.230+03	.234+02				
.240+03	.233+02				
.250+03	.233+02				
.260+03	.228+02				
.270+03	.204+02				
.280+03	.166+02				
.290+03	.112+02				
.300+03	.679+01				
.310+03	.436+01				
.320+03	.341+01				
.330+03	.310+01				
.340+03	.299+01				
.350+03	.299+01				
.360+03	.239+01				
.370+03	.299+01				
.330+03	.239+01				

@FIN

PROGRAM LISTING

(Pages 105 to 118)

```
SELT.ILD
ELTOT7 RLI37C 03/25-14:51:07
 000          COMMON AKOR,AKOL,AKOW,PI,CNVA,DNM,LIN,CH1,CH2,YNM,ATPC,CONWA,
 000         1 DELR,DELI,DELA
 000          INTEGER READER
 000          DOUBLE PRECISION DEI,ARG1,ARG2,DI(55),CH1,CH2,XC,AGRC,ATPC,ATROP
 000          DOUBLE PRECISION AGRC2
 000   C      DIMENSION OF DI         MUST BE AT LEAST MAX(X,N)+20 OR
 000   C      MAX(2*X,N) FOR X>20
 000          DIMENSION VAR(2),X(53)
 000          DIMENSION PLTT(101),PLTR(101),PLTX(21),PLTC(21),PLTK(21),
 000         1 PLTE(21,5),PLTS(21,5),PLTD(21,5),LBX(5),LBY(3),LBXX(2),LBYE(3),
 000         2 LBYD(2),LBYP(2),LBYO(2),ICHP(5),FFM(5)
 000     1000 FORMAT( )
 000     1011 FORMAT (5H NFM=,I3,3X,7H NFMPD=,I3)
 000     1012 FORMAT (13H PATTERN SLS=,E10.3,3X,7H FC/F-=,E10.3,3X,4H DB=,E10.3,
 000         1 3X,4H NP=,I3,3X,4H TD=,E10.7)
 000     1013 FORMAT (1H1,7X,4H IOP,8X,4H IOE,8X,4H IOS,8X,4H IOD,8X,4H IOC)
 000     1014 FORMAT (5(4X,I3,5X))
 000     1015 FORMAT (3X,7H FC/FM=,5(2X,E10.3))
 000     1016 FORMAT (3X,6H ALTRC,6X,5H P1P2)
 000     1017 FORMAT (2(E10.3,3X))
 000     1018 FORMAT (5H LIN=,I2,3X,5H IOC=,I2)
 000     1019 FORMAT (6H KO*A=,E10.3)
 000     1020 FORMAT (6H KO*L=,E10.3,3X,6H KO*W=,E10.3)
 000     1030 FORMAT (8H FC/F- =,E10.3,3X,4H CN=,E10.3)
 000     1040 FORMAT (22H NFM MUST BE 5 OR LESS)
 000     1050 FORMAT (1H1,4X,5H SLC*,7X,8H SL-(1)*,4X,8H SL-(2)*,4X,8H SL-(3)*,
 000         1 4X,8H SL-(4)*,4X,8H SL-(5)*)
 000     1051 FORMAT (1H1,4X,5H SLC*,7X,8H SL+(1)*,4X,8H SL+(2)*,4X,8H SL+(3)*,
 000         1 4X,8H SL+(4)*,4X,8H SL+(5)*)
 000     1060 FORMAT (1X,6(2X,E10.3))
 000     1070 FORMAT (////)
 000     1080 FORMAT (10H CPU TIME=,I4,8H SECONDS)
 000     1100 FORMAT (13H ALPHA(T)*RO=,E10.3,3H DB)
 000     1110 FORMAT (7H CURVE ,I2,14H IS FOR FC/F-=,E10.3)
 000     1120 FORMAT (43H PRESSURE UNITS DB/1 MICRO PASCAL (KHZ) RMS)
 000     1130 FORMAT (13H ALPHA(O)*RO=,E10.3,7H NEPERS)
 000     1140 FORMAT (6H SLE*=,E10.3,3X,7H FC/F-=,E10.3)
 000     1150 FORMAT (5H NOC=,I3)
 000     1260 FORMAT (1H1,4X,5H SLC*,7X,7H SL OUT)
 000     1270 FORMAT (10X,37H ***** F1+F2 UP CONVERSION CASE *****///)
 000     1280 FORMAT (1H1,4X,5H SLC*,7X,6H DI(1),6X,6H DI(2),6X,6H DI(3),6X,
 000         1 6H DI(4),6X,6H DI(5))
 000          READER=5
 000   C      UNSCALED VALUES OF SPL- AND SPLC CAN BE OBTAINED BY SPECIFYING
 000   C      KOA AND USING THE FOLLOWING EQUATIONS
 000   C      SPLC    =SPLC*    -40.0*LOG10(KOA)+18.46
 000   C      SPL-    =SPL-*    -40.0*LOG10(KOA)+18.46
 000   C      NOTE THAT THE CONSTANT 18.46 DB ONLY PERTAINS TO A WATER MEDIUM,
 000   C      IN A MEDIUM WITH A DIFFERENT SOUND SPEED THIS CONSTANT CAN BE
 000   C      CHANGED BY ADDING 20.0*LOG10(1.5/(C OF THE NEW MEDIUM IN
 000   C      KILOMETERS/SEC)) TO 18.46
 000   C      CONWA=MEDIUM CONSTANT=290.0 FOR WATER
 000          CONWA=290.0
 000          READ(READER,1000) NOC
```

```
      PRINT 1150,NCC
      INCC=0
    5 CONTINUE
      INCC=INCC+1
      READ(READER,1000) ICP,ICE,ICS,ICD,ICC
      PRINT 1013
      PRINT 1014,ICP,ICE,ICS,ICD,ICC
      NS=0
      IC=ICE+ICS+ICD+ICC
      IF (IC .GT. 0) NS=21
      READ(READER,1000) ALTRC,P1P2
      ATRC=ALTRC/8.67
C     ALTRC IS IN DB, ATRC IS IN NEPERS
C     A NEGATIVE ALTRC IS FOR F1*F2
      PRINT 1016
      PRINT 1017,ALTRC,P1P2
      READ(READER,1000) NFM,NFMRD
      PRINT 1011,NFM,NFMRD
      IF (NFMRD .NE. 1) GO TO 10
      IF (NFM .GT. 5) GO TO 13
      READ(READER,1000) (FFM(K),K=1,NFM)
C     FFM=F0/F-
      PRINT 1015, (FFM(K),K=1,NFM)
      GO TO 10
   13 PRINT 1040
      GO TO 990
   10 CONTINUE
      IF (ICP .EQ. 0 .AND. ICD .EQ. 0) GO TO 16
      READ(READER,1000) LIN,ICC
      PRINT 1018,LIN,ICC
C     LIN=0 FOR PISTON,1 FOR RECTANGLE
C     ICC=1 FOR F(THETA), =2 FOR DELTA(THETA), =3 FOR CONVOLUTION
      IF (LIN .EQ. 1) GO TO 11
      READ(READER,1000) AKCR
C     AKCR=KC*A=2.0*PI*(PISTON RADIUS)/WAVELENGTH
      PRINT 1019, AKCR
      GO TO 12
   11 READ(READER,1000) AKCL,AKCW
C     AKCL=2.0*PI*(RECTANGLE LENGTH)/WAVELENGTH
C     AKCW=2.0*PI*(RECTANGLE WIDTH)/WAVELENGTH
      PRINT 1020, AKCL,AKCW
   12 CONTINUE
      IF (ICP .NE. 1) GO TO 16
      READ(READER,1000) SLS,PFFM,DB,NP,TC
      PRINT 1012,SLS,PFFM,DB,NP,TC
      IF (DB .GT. 0.0) DB=-DB
   16 CONTINUE
      PRINT 1070
      IF (ALTRC .LT. 0.0) PRINT 1270
      PI=3.14159
      ICHR(1)='1'
      ICHR(2)='2'
      ICHR(3)='3'
      ICHR(4)='4'
      ICHR(5)='5'
      LSXX(1)=4
```

```
      LBXX(2)='SLO.'
      LEYS(1)=1C
      LEYS(2)='2C LOC'
      LEYS(3)=' ETA'
      LBYS(1)=4
      LBYS(2)='SL-.'
      IF (ALTRC .LT. 0.0) LEYS(2)='SL+.'
      LBYD(1)=2
      LEYD(2)='DI'
      LBYD(1)=6
      LBYD(2)='SL OUT'
C     ABRAMOWITZ PAGE 231 FORMULA 5.1.56 FOR EXP(X)*E1(X)
      A1=4.03640
      A2=1.15198
      B1=5.03637
      B2=4.19160
      IF (ATRC .GE. 10.000) GO TO 61
      DEL=-DEXP(ATRC)*DEI(-ATRC)
      GO TO 62
   61 DEL=(ATRC**2+A1*ATRC+A2)/(ATRC**3+B1*ATRC**2+B2*ATRC)
   62 DELL=DEL
      DLLL=DELL
      IF (ALTRC .LT. 0.0) DLLL=-DEXP(DABS(ATRC))*DEI(-DABS(ATRC))
      DO 300 K=1,NFM
      IF (NFMRD .EQ. 1) GO TO 14
      FCFM=5.0*(2.0**(K-1))
      FFM(K)=FCFM
      GO TO 15
   14 FCFM=FFM(K)
   15 CONTINUE
      F1FC=1.0+0.5/FCFM
      F2FC=1.0-0.5/FCFM
      F1F2=F1FC/F2FC
      IF (ALTRC .LT. 0.0) GO TO 67
      A1RC=ATRC/2.0*F1F2
      A2RC=ATRC/(2.0*F1F2)
      ACRC=(A1RC+A2RC+ATRC)/4.0
      GO TO 68
   67 A1RC=-ATRC/2.0*F1F2
      A2RC=-ATRC/(2.0*F1F2)
      ACRC=(A1RC+A2RC-ATRC)/4.0
   68 CONTINUE
      ACRC2=2.0*ACRC
      IF (ACRC .GE. 10.000) GO TO 63
      DELC=-DEXP(ACRC2)*DEI(-ACRC2)
      GO TO 64
   63 DELC=(ACRC2**2+A1*ACRC2+A2)/(ACRC2**3+B1*ACRC2**2+B2*ACRC2)
   64 CONTINUE
      ATRCP=DABS(ATRC)*FCFM
      IF (ATRCP .GE. 10.000) GO TO 65
      DELP=-DEXP(ATRCP)*DEI(-ATRCP)
      GO TO 66
   65 DELP=(ATRCP**2+A1*ATRCP+A2)/(ATRCP**3+B1*ATRCP**2+B2*ATRCP)
   66 CONTINUE
      EN=1.0+DLOG10(DLLL/DELP)/DLOG10(FCFM)
      PRINT 1030,FCFM,EN
```

```
      IF (NS .EQ. 0) GO TO 800
      DO 801 J=1,NS
      SLOS=180.0+10.0*(J-1)
      PLTX(J)=SLOS
      SL1S=SLOS-10.0*ALOG10(1.0+(1.0/P1P2)**2)
      SL2S=SLOS-10.0*ALOG10(1.0+P1P2**2)
      XODB=SLOS+20.0*ALOG10(DELO)-CONWA
      CH1DB=SL1S+20.0*ALOG10(ABS(DEL1))-CONWA
      CH2DB=SL2S+20.0*ALOG10(ABS(DEL1))-CONWA
      CHODB=SLOS+20.0*ALOG10(ABS(DEL1))-CONWA
      XO=10.0**(XODB/20.0)
      CH1=10.0**(CH1DB/20.0)
      CH2=10.0**(CH2DB/20.0)
      CHO=10.0**(CHODB/20.0)
      IF (XO .GE. 73.000) GO TO 20
      CALL BESI(XO,0.00,1,BI,65)
      FNO=BI(2)/BI(1)
      GO TO 21
   20 FNO=1.0
   21 CONTINUE
      SLOUT=(CONWA+6.0)-20.0*ALOG10(DELO/FNO)
      IF (IOD .EQ. 1) PLTO(J)=SLOUT
      IF (CH1 .GE. 33.000) GO TO 22
      CALL BESI(CH1,0.00,1,BI,65)
      FNI=BI(2)/BI(1)
      GO TO 23
   22 FNI=1.0
   23 CONTINUE
      IF (CH2 .GE. 33.000) GO TO 24
      CALL BESI(CH2,0.00,1,BI,65)
      FMI=BI(2)/BI(1)
      GO TO 25
   24 FMI=1.0
   25 CONTINUE
      FCHI=2.0*FNI*FMI/CHO
      IF (ALTRO .LT. 0.0) GO TO 71
      ETADB=20.0*ALOG10(FCHI)-20.0*EN*ALOG10(FCFM)
      GO TO 72
   71 ETADB=20.0*ALOG10(FCHI)-20.0*EN*ALOG10(0.5)
   72 CONTINUE
      IF (IOD .EQ. 1) PLTE(J,K)=ETADB
      SLMS=SLOS+ETADB
      IF (IOD .EQ. 1) PLTS(J,K)=SLMS
      IF (IOD .NE. 1) GO TO 40
C     DI SECTION - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (IOD .NE. 3) GO TO 90
      VAR(1)=0.0
      VAR(2)=0.0
      CALL CNFAT(VAR,SLOS,FCFM)
      CNVX=CNVA
   90 CONTINUE
      IF (IOD .EQ. 2) GO TO 30
      ARG1=CH1*F1FC
      ARG2=CH2*F2FC
      IF (ARG1 .GE. 33.000) GO TO 26
      CALL BESI(ARG1,0.00,1,BI,65)
```

```
000223    000          YN=BI(2)/BI(1)
000224    000          GO TO 27
000225    000       26 YN=1.0
000226    000       27 CONTINUE
000227    000          IF (ARG2 .GE. 73.000) GO TO 28
000228    000          CALL BESI(ARG2,0.00,1,BI,65)
000229    000          YM=BI(2)/BI(1)
000230    000          GO TO 29
000231    000       28 YM=1.0
000232    000       29 CONTINUE
000233    000          YNM=YN*YM
000234    000       30 CONTINUE
000235    000          IF (LIN .EQ. 1 .AND. IOC .NE. 2) GO TO 750
000236    000          TA=0.0
000237    000          TB=PI
000238    000          HSTAR=PI*1.0E-2
000239    000          HMIN=PI*1.0E-4
000240    000          HMAX=PI/2.0
000241    000          ERMAX=1.0E-3
000242    000          KEY=0
000243    000    C     FOR PISTON, PATTERN ASSUMED ZERO BEHIND PISTON
000244    000          CALL ROM2S(TA,TB,THET,FOFX,HSTAR,HMIN,HMAX,ERMAX,ANS,KSTOP,KEY)
000245    000      700 IF (THET .GT. PI/2.0) GO TO 745
000246    000          VAR(1)=0.0
000247    000          VAR(2)=THET
000248    000          GO TO (710,720,730),IOC
000249    000      710 CALL SMPAT(VAR,SLOS,FOFM)
000250    000          FOFX=ONM**2 *SIN(THET)
000251    000          GO TO 746
000252    000      720 CALL DLPAT(VAR,SLOS,FOFM)
000253    000          FOFX=DELA**2 *SIN(THET)
000254    000          GO TO 746
000255    000      730 CALL CNPAT(VAR,SLOS,FOFM)
000256    000          FOFX=(CNVA/CNVX)**2 *SIN(THET)
000257    000          GO TO 746
000258    000      745 FOFX=0.0
000259    000      746 CONTINUE
000260    000          CALL ROM2
000261    000          IF (KSTOP .EQ. 1) GO TO 700
000262    000          DI=10.0*ALOG10(4.0*PI/(2.0*PI*ANS))
000263    000          PLTD(J,K)=DI
000264    000          GO TO 301
000265    000      750 NINT=2
000266    000          ERMAX=1.0E-3
000267    000          KEY=0
000268    000          CALL RM01(NINT,VAR,FUN,ANS,ERMAX,KOO,WK,KEY)
000269    000      751 TA1=0.0
000270    000          TB1=2.0*PI
000271    000          HSTAR1=PI*1.0E-2
000272    000          HMIN1=PI*1.0E-4
000273    000          HMAX1=PI/2.0
000274    000          CALL RM02(TA1,TB1,HSTAR1,HMIN1,HMAX1)
000275    000      752 CALL RM02(0.0,PI,0.01*PI,PI*1.0E-4,PI/2.0)
000276    000      753 IF (IOC .EQ. 3) GO TO 760
000277    000          CALL SMPAT(VAR,SLOS,FOFM)
000278    000          FUN=ONM**2 *SIN(VAR(2))
```

```
000          GO TO 761
000      760 CALL CNRAT(VAR,SLOS,FSFM)
000          FUN=(CNVA/CNVX)**2 *SIN(VAR(2))
000      761 CALL RM33
000          GO TO (751,752,753,754),KGO
000      754 DI=10.0*ALOG10(4.0*PI/ANS)
000          PLTS(J,K)=DI
000      751 CONTINUE
000    C     END OF DI SECTION - - - - - - - - - - - - - - - - - - - - - - - - - - -
000       40 CONTINUE
000      801 CONTINUE
000      800 CONTINUE
000          IF (ICE .NE. 1) GO TO 810
000          DO 802 K=1,NFM
000          DO 803 J=1,NS
000          PLTK(J)=PLTE(J,K)
000      803 CONTINUE
000          IF (K .GT. 1) GO TO 804
000          CALL PSCALE(PLTK,NS,YLO,YHI,.TRUE.,1)
000          GO TO 802
000      804 CALL PSCALE(PLTK,NS,YLO,YHI,.FALSE.,1)
000      802 CONTINUE
000          CALL PSETUP(180.,380.,YLO,YHI,1HF,1H ,LBXX,LBYE)
000          DO 805 K=1,NFM
000          DO 806 J=1,NS
000          PLTK(J)=PLTE(J,K)
000      806 CONTINUE
000          IF (K .EQ. NFM) GO TO 807
000          CALL PPLOT(.FALSE.,ICHR(K),NS,PLTX,PLTK)
000          GO TO 805
000      807 CALL PPLOT(.TRUE.,ICHR(K),NS,PLTX,PLTK)
000      805 CONTINUE
000          PRINT 1100,ALTRC
000          DO 850 K=1,NFM
000      850 PRINT 1110,K,FFM(K)
000          PRINT 1120
000      810 CONTINUE
000          IF (ICS .NE. 1) GO TO 820
000          CALL PSETUP(180.,380.,0.,320.,1HF,1H ,LBXX,LBYS)
000          DO 815 K=1,NFM
000          DO 816 J=1,NS
000          PLTK(J)=PLTS(J,K)
000      816 CONTINUE
000          IF (K .EQ. NFM) GO TO 817
000          CALL PPLOT(.FALSE.,ICHR(K),NS,PLTX,PLTK)
000          GO TO 815
000      817 CALL PPLOT(.TRUE.,ICHR(K),NS,PLTX,PLTK)
000      815 CONTINUE
000          PRINT 1100,ALTRC
000          DO 851 K=1,NFM
000      851 PRINT 1110,K,FFM(K)
000          PRINT 1120
000          IF (ALTRC .LT. 0.0) GO TO 811
000          PRINT 1050
000          GO TO 812
000      811 PRINT 1051
```

```
000      815 CONTINUE
000          DO 813 J=1,NS
000          PRINT 1260,PLTX(J),(PLTS(J,K),K=1,NFM)
000      813 CONTINUE
000      820 CONTINUE
000          IF (ICD .NE. 1) GO TO 830
000          DO 822 K=1,NFM
000          DO 823 J=1,NS
000          PLTK(J)=PLTD(J,K)
000      823 CONTINUE
000          IF (K .GT. 1) GO TO 824
000          CALL PSCALE(PLTK,NS,YLO,YHI,.TRUE.,1)
000          GO TO 822
000      824 CALL PSCALE(PLTK,NS,YLO,YHI,.FALSE.,1)
000      822 CONTINUE
000          CALL PSETUP(130.,330.,YLO,YHI,1HF,1H ,LBYX,LBYD)
000          DO 825 K=1,NFM
000          DO 826 J=1,NS
000          PLTK(J)=PLTD(J,K)
000      826 CONTINUE
000          IF (K .EQ. NFM) GO TO 827
000          CALL PPLOT(.FALSE.,ICHR(K),NS,PLTX,PLTK)
000          GO TO 825
000      827 CALL PPLOT(.TRUE.,ICHR(K),NS,PLTX,PLTK)
000      825 CONTINUE
000          PRINT 1100,ALTPC
000          DO 852 K=1,NFM
000      852 PRINT 1110,K,FFM(K)
000          PRINT 1120
000          PRINT 1260
000          DO 826 J=1,NS
000          PRINT 1260,PLTX(J),(PLTD(J,K),K=1,NFM)
000      826 CONTINUE
000      830 CONTINUE
000          IF (ICC .NE. 1) GO TO 840
000          CALL PSCALE(PLTC,NS,YLO,YHI,.TRUE.,1)
000          CALL PSETUP(130.,330.,YLO,YHI,1HF,1H ,LBYX,LBYC)
000          CALL PPLOT(.TRUE.,1H*,NS,PLTX,PLTC)
000          PRINT 1130,ACRD
000          PRINT 1120
000          PRINT 1280
000          DO 831 J=1,NS
000      831 PRINT 1280,PLTX(J),PLTC(J)
000      840 CONTINUE
000          IF (ICF .NE. 1) GO TO 860
000          IF (ICC .EQ. 3) GO TO 130
000          SL1S=SLS -10.0*ALOG10(1.0+(1.0/P1P2)**2)
000          SL2S=SLS -10.0*ALOG10(1.0+P1P2**2)
000          CH1DB=SL1S+20.0*ALOG10(ABS(DELL))-CONWA
000          CH2DB=SL2S+20.0*ALOG10(ABS(DELL))-CONWA
000          CH1=10.0**(CH1DB/20.0)
000          CH2=10.0**(CH2DB/20.0)
000          ARG1=CH1*F1FP
000          ARG2=CH2*F2FP
000          IF (ARG1 .GE. 33.000) GO TO 120
000          CALL SESI(ARG1,0.00,1,BI,65)
```

```
      YN=SI(2)/DI(1)
      GO TO 127
  126 YN=1.0
  127 CONTINUE
      IF (ARG2 .GE. 33.000) GO TO 128
      CALL SEDI(ARG2,0.00,1,DI,65)
      YM=SI(2)/DI(2)
      GO TO 129
  128 YM=1.0
  129 CONTINUE
      YNM=YN*YM
  133 CONTINUE
      IF (ICC .NE. 3) GO TO 190
      VAR(1)=0.0
      VAR(2)=0.0
      CALL CNPAT(VAR,SLS,PFFM)
      CNVX=CNVA
  190 CONTINUE
      DO 300 I=1,NP
      THET= TD*I*PI/180.0
      THETD= TD*I
      PLTT(I)=THETD
      VAR(1)=0.0
      VAR(2)=THET
C     PATTERN ASSUMED IN PLANE OF LENGTH AXIS FOR RECTANGLE
      GO TO (852,853,854),ICC
  852 CALL EMPAT(VAR,SLS,PFFM)
      PLTP(I)=20.0*ALOG10(ABS(DNM))
      GO TO 300
  853 CALL DLPAT(VAR,SLS,PFFM)
      PLTP(I)=20.0*ALOG10(DELA)
      GO TO 300
  854 CALL CNPAT(VAR,SLS,PFFM)
      PLTP(I)=20.0*ALOG10(CNVA/CNVX)
      CALL SECOND(ISEC)
      PRINT 1090,ISEC
  300 CONTINUE
      LBX(1)=23
      LBX(2)='PATTER'
      LBX(3)='N ANGL'
      LBX(4)='E (DEG'
      LBX(5)='REES)'
      LBY(1)=11
      LBY(2)='PRESSU'
      LBY(3)='RE DB'
      TF=100.0*TD
      CALL PSETUP(0.0,TF,20,0.0,1HF,1H ,LBX,LBY)
      CALL PPLOT(.TRUE.,1H*,NP,PLTT,PLTP)
      PRINT 1140,SLS,PFFM
      PRINT 1100,ALTRD
      PRINT 1120
  360 CONTINUE
  990 IF (INCC .LT. NCC) GO TO 5
  999 END

END ELT.
```

```
BELT.ILS
ELTOT7 RL1372 09/25-14:51:11
       000         SUBROUTINE IMPAT(VAR,SL,FF)
       000         COMMON AKZP,AKCL,AKCW,PI,CNVR,ENM,LIN,CH1,CH2,YNM,ATPC,CONWA,
       000        1 DELP,DELI,DELA
       000         DOUBLE PRECISION ARG1,ARG2,TERM1,TERM2,CH1,CH2,B0(2),B1(2),B2(65)
       000         DIMENSION VAR(2)
       000    C    PATTERN CALCULATION - - - - - - - - - - - - - - - - - - - - - - - -
       000         THET=VAR(2)
       000         F1FD=1.0+0.5/FF
       000         F2FD=1.0-0.5/FF
       000         IF (LIN .EQ. 1) GO TO 100
       000         TERM1=DABS(AKCP*F1FD*SIN(THET))
       000         TERM2=DABS(AKCP*F2FD*SIN(THET))
       000         IF (TERM1 .EQ. 0.00) GO TO 710
       000         CALL BJYC1(TERM1,2,1,B0,B1)
       000         D1=2.0*B1(1)/TERM1
       000         GO TO 720
       000     710 D1=1.0
       000     720 IF (TERM2 .EQ. 0.00) GO TO 730
       000         CALL BJYC1(TERM2,2,1,B0,B1)
       000         D2=2.0*B1(1)/TERM2
       000         GO TO 771
       000     730 D2=1.0
       000         GO TO 771
       000     100 CX=SIN(VAR(2))*COS(VAR(1))
       000         CY=SIN(VAR(2))*SIN(VAR(1))
       000         CZ=COS(VAR(2))
       000         IF (CX .EQ. 0.0) GO TO 761
       000         TNL=SQRT(CZ**2+CY**2)/CX
       000         THL=PI/2.0-ABS(ATAN(TNL))
       000         GO TO 762
       000     761 THL=0.0
       000     762 IF (CY .EQ. 0.0) GO TO 763
       000         TNW=SQRT(CZ**2+CX**2)/CY
       000         THW=PI/2.0-ABS(ATAN(TNW))
       000         GO TO 764
       000     763 THW=0.0
       000     764 ANGL1=(AKCL*F1FD/2.0)*SIN(THL)
       000         ANGL2=(AKCL*F2FD/2.0)*SIN(THL)
       000         ANGW1=(AKCW*F1FD/2.0)*SIN(THW)
       000         ANGW2=(AKCW*F2FD/2.0)*SIN(THW)
       000         IF (ANGL1 .EQ. 0.0) GO TO 765
       000         IF (ANGW1 .EQ. 0.0) GO TO 766
       000         D1=SIN(ANGL1)/ANGL1*SIN(ANGW1)/ANGW1
       000         GO TO 767
       000     766 D1=SIN(ANGL1)/ANGL1
       000         GO TO 767
       000     765 IF (ANGW1 .EQ. 0.0) GO TO 768
       000         D1=SIN(ANGW1)/ANGW1
       000         GO TO 767
       000     768 D1=1.0
       000     767 IF (ANGL2 .EQ. 0.0) GO TO 769
       000         IF (ANGW2 .EQ. 0.0) GO TO 770
       000         D2=SIN(ANGL2)/ANGL2*SIN(ANGW2)/ANGW2
       000         GO TO 771
```

```
000   770 D2=SIN(ANGL2)/ANGL2
000       GO TO 771
000   769 IF (ANGW2 .EQ. 0.0) GO TO 772
000       D2=SIN(ANGW2)/ANGW2
000       GO TO 771
000   772 D2=1.0
000   771 CONTINUE
000       ARG1=CM1*D1*F1FC
000       ARG2=CM2*D2*F2FC
000       ARG1=DABS(ARG1)
000       ARG2=DABS(ARG2)
000       IF (ARG1 .GE. 33.0D0) GO TO 773
000       CALL BESI(ARG1,0.0D0,1,BI,65)
000       DN=BI(2)/BI(1)
000       GO TO 774
000   773 DN=1.0
000   774 CONTINUE
000       IF (D1 .LT. 0.0) DN=DN*(-1.0)**1
000       IF (ARG2 .GE. 33.0D0) GO TO 775
000       CALL BESI(ARG2,0.0D0,1,BI,65)
000       DM=BI(2)/BI(1)
000       GO TO 776
000   775 DM=1.0
000   776 CONTINUE
000       IF (D2 .LT. 0.0) DM=DM*(-1.0)**1
000       DNM=DN*DM/YNM
000       GO TO 200
000   200 RETURN
000       END

END ELT.
```

```
SELT,ILD
ELT0T7 RL1370 09/25-14:51:11
        000          SUBROUTINE DLPAT(VAR,DL,FF)
        000          COMMON AKOR,AKOL,AKOW,PI,CNV1,DNM,LIN,CH1,CH2,YNM,ATRC,CONWA,
        000        1 DELR,DELI,DELA
        000          DOUBLE PRECISION CH1,CH2,ATRC
        000          DIMENSION VAR(2)
        000          THET=VAR(2)
        000          INF=20
        000          DODB=DL  -CONWA-20.0*ALOG10(ABS(ATRC/2.0))
        000          XK=(10.0**(DODB/20.0))/4.0
        000          D=ATAN(1.0/XK)
        000          SQ1PX2=SQRT(1.0+XK**2)
        000          SQ1OX2=SQRT(1.0+(1.0/XK)**2)
        000          IF (LIN .EQ. 1) GO TO 13
        000          AKMRC=(AKOR**2)/(2.0*FF)
        000          IF (ATRC .LT. 0.00) AKMRC=AKOR**2
        000          GO TO 14
        000       13 AKMRC=AKOL*AKOW/(2.0*PI*FF)
        000          IF (ATRC .LT. 0.00) AKMRC=AKOL*AKOW/PI
        000       14 CONTINUE
        000          CF=2.0*AKMRC/ATRC
        000          Y=CF*(SIN(THET/2.0))**2
        000          SMR1=0.0
        000          SMI1=0.0
        000          SMX1=0.0
        000          DO 10 N=1,INF
        000          IF (XK .LT. 1.0) GO TO 11
        000          AN=SIN(N*D)/(XK*(SQ1OX2)**N)
        000          GO TO 12
        000       11 AN=SIN(N*D)*((XK)**(N-1))/(SQ1PX2**N)
        000       12 CONTINUE
        000          TEMP1=AN/N
        000          TEMP2=1.0+(Y/N)**2
        000          SMX1=SMX1+TEMP1
        000          SMR1=SMR1+TEMP1/TEMP2
        000          SMI1=SMI1-TEMP1*(Y/N)/TEMP2
        000       10 CONTINUE
        000          DELR=SMR1/SMX1
        000          DELI=SMI1/SMX1
        000          DELA=SQRT(DELR**2+DELI**2)
        000          RETURN
        000          END

END ELT.
```

```
SELT.ILC
ELT077 RL1076 03/25-14:51:12
      SUBROUTINE CMPAT(VAR,SL,FF)
      COMMON AKCP,AKCL,AKCW,PI,CNVA,DNM,LIN,CH1,CH2,YNM,ATRC,CONWA,
     1 DELP,DELI,DELA
      DOUBLE PRECISION CH1,CH2,ATRC
      DIMENSION VAR(2),WK(53),VP(2),VPI(2)
      IF (LIN .EQ. 1) GO TO 200
      TA=-PI/2.0
      TB=PI/2.0
      HSTAR=PI*1.0E-2
      HMIN=PI*1.0E-4
      HMAX=PI/2.0
      ERMAX=1.0E-3
      KEY=0
      CALL ROMBS(TA,TB,THETP,FOFX,HSTAR,HMIN,HMAX,ERMAX,ANS,KSTOP,KEY)
  100 VP(1)=0.0
      VP(2)=THETP
      CALL DMPAT(VP,SL,FF)
      VP(1)=0.0
      VP(2)=VAR(2)-THETP
      CALL DLPAT(VP,SL,FF)
      FOFX=DNM*DELP*COS(THETP)
      CALL ROM2
      IF (KSTOP .EQ. 1) GO TO 100
      CNVR=2.0*PI*ANS
      CALL ROMBS(TA,TB,THETP,FOFX,HSTAR,HMIN,HMAX,ERMAX,ANS,KSTOP,KEY)
  110 VP(1)=0.0
      VP(2)=THETP
      CALL BMPAT(VP,SL,FF)
      VP(1)=0.0
      VP(2)=VAR(2)-THETP
      CALL DLPAT(VP,SL,FF)
      FOFX=DNM*DELI*COS(THETP)
      CALL ROM2
      IF (KSTOP .EQ. 1) GO TO 110
      CNVI=2.0*PI*ANS
      CNVA=SQRT(CNVR**2+CNVI**2)
      GO TO 300
  200 NINT=2
      ERMAX=1.0E-3
      KEY=0
      CALL RME1(NINT,VP ,FUN,ANS,ERMAX,KSC,WK,KEY)
  210 TA1=0.0
      TB1=2.0*PI
      HSTAR1=PI*1.0E-2
      HMIN1=PI*1.0E-4
      HMAX1=PI/2.0
      CALL RMI2(TA1,TB1,HSTAR1,HMIN1,HMAX1)
  220 TA2=-PI/2.0
      TB2=PI/2.0
      CALL RMI2(TA2,TB2,0.01*PI,PI*1.0E-4,PI/2.0)
  230 CALL BMPAT(VP,SL,FF)
      VP1(1)=0.0
      VP1(2)=VAR(2)-VP(2)
      CALL DLPAT(VP1,SL,FF)
```

```
000      FUN=DNM*DELR*COS(VP(2))
000      CALL RM03
000      GO TO (210,220,230,240),KGO
000  240 CNVR=ANS
000      CALL RM01(NINT,VP ,FUN,ANS,EPMAX,KGO,WK,KEY)
000  250 CALL RM02(TA1,TB1,HSTAR1,HMIN1,HMAX1)
000  260 CALL RM02(TA2,TB2,0.01,PI,PI,1.0E-4,PI/2.0)
000  270 CALL DMPAT(VP,SL,FF)
000      VP1(1)=0.0
000      VP1(2)=VAR(2)-VP(2)
000      CALL DLPAT(VP1,SL,FF)
000      FUN=DNM*DELI*COS(VP(2))
000      CALL RM03
000      GO TO (250,260,270,280),KGO
000  280 CNVI=ANS
000      CNVA=SQRT(CNVR**2+CNVI**2)
000  300 RETURN
000      END

END ELT.

@FIN
```